

UNIVERSITÉ DE MONTRÉAL

2D AND 3D VERIFICATION AND VALIDATION OF THE LATTICE BOLTZMANN
METHOD

MATTEO PORTINARI
DÉPARTEMENT DE GÉNIE MÉCANIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE MÉCANIQUE)
SEPTEMBRE 2015

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

2D AND 3D VERIFICATION AND VALIDATION OF THE LATTICE BOLTZMANN
METHOD

présenté par : PORTINARI Matteo

en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de :

M. CAMARERO Ricardo, Ph. D., président

M. TRÉPANIÉ Jean-Yves, Ph. D., membre et directeur de recherche

M. REGGIO Marcelo, Ph. D., membre et codirecteur de recherche

M. VÉTEL Jérôme, Doctorat, membre

DEDICATION

To my parents

ACKNOWLEDGMENTS

Firstly, I would like to thank my research supervisor, Professor Jean-Yves Trépanier, for the freedom and trust he granted me throughout my research. I am grateful for the very useful and pertinent advice he provided me during my work and for the quick and precise revision of the hereby thesis.

I also express my gratitude towards my research co-supervisor, Professor Marcelo Reggio, for his guidance, support and for being the first to introduce me to the Lattice Boltzmann Method.

I would also like to thank my friends and co-workers for the two years we spent together. I particularly appreciated the exchanges of opinions and the friendly atmosphere in our lab. A special thought goes to Sibylle, for her unyielding support and encouragements.

Lastly, I would like to express my gratitude towards my parents for having been close to me, despite the geographic distance, and for the trust they put in me all along my studies. A special "thank you" goes to my grand-parents and to my family all for the support, food and accommodation they provided me in these years as "errant" student.

REMERCIEMENTS

Je tiens à remercier mon directeur de recherche, le Professeur Jean-Yves Trépanier, pour la liberté et la confiance qu'il m'a accordées pendant ma recherche. Je vous remercie pour les conseils toujours utiles que vous m'avez donnés tout au long de mon travail ainsi que pour la correction précise et rapide de ce mémoire.

J'exprime également ma gratitude envers mon codirecteur de recherche, le Professeur Marcelo Reggio, pour ses conseils et ses opinions et pour m'avoir fait découvrir en premier la méthode de Boltzmann sur réseau.

Je remercie aussi tous mes collègues de travail pour les discussions précieuses et pour l'ambiance amicale qui s'est créée entre nous pendant ces années. Une pensée particulière pour Sibylle, pour son support moral et ses encouragements.

Infine, vorrei ringraziare i miei genitori per essermi stati vicini, nonostante la lontananza, e per la fiducia accordatami durante i miei studi. Un grazie particolare va anche ai miei nonni e alla mia famiglia, per il supporto, il cibo e l'ospitalità che mi avete offerto durante questi anni da studente "errante".

RÉSUMÉ

Durant ces dernières années, la Méthode de Boltzmann sur Réseau (LBM) a gagné beaucoup d'intérêt pour les simulations d'écoulements en dynamique des fluides. Dans ce contexte, le but de ce mémoire est de développer un code LBM pour ensuite le vérifier et le valider pour différents cas de référence, afin d'explorer ses capacités dans plusieurs situations variées et diverses.

Le travail est constitué de quatre parties. Premièrement, une revue de la littérature fournit des informations au sujet de l'origine et de l'état de l'art en ce qui concerne le LBM. L'attention est posée en particulier sur les deux procédés qui constituent le coeur de la méthode, notamment collision et propagation. Ils sont décrits avec leurs équations associées. L'accent est aussi mis sur la modélisation de la turbulence avec le LBM.

Dans la deuxième partie, les différentes étapes qui caractérisent le développement d'un code LBM sont identifiées et détaillées. En particulier, il y a une focalisation majeure sur la conversion entre un domaine physique et le réseau ainsi que sur l'implémentation des différentes conditions limite. Ensuite est introduit le modèle de collision à temps de relaxation multiples (MRT), qui permet une amélioration en terme de stabilité.

La troisième partie présente les résultats issus du code LBM à deux dimensions. D'abord, un écoulement de Poiseuille est simulé et les résultats sont comparés à la solution analytique correspondante. Deux conditions limite différentes sont comparées en terme de précision et une étude sur l'ordre de convergence de la méthode LBM est effectuée. Ensuite, l'écoulement de Couette-Poiseuille ainsi que l'écoulement dans une cavité entraînée sont implémentés. Les résultats sont comparés avec leurs correspondants, issus respectivement d'une solution analytique et d'une solution expérimentale. L'écoulement instationnaire autour d'un obstacle est aussi simulé et validé par comparaison avec les résultats obtenus avec une méthode à volumes finis.

La quatrième et dernière partie est dédiée aux simulations tri-dimensionnelles. Initialement, les changements liés à un passage de deux à trois dimensions sont décrits. Ensuite, les implémentations de deux écoulements avec gradient de pression sont étudiées et comparées aux solutions analytiques correspondantes. Une étude sur l'ordre de convergence en trois dimensions est aussi présentée. Successivement, le défi de la modélisation de la turbulence dans le LBM est introduit. Les principaux aspects théoriques du modèle de turbulence LES (Large Eddy Simulation) sont illustrés ainsi que les équations liées. Pour conclure, les résultats issus d'une simulation LBM d'un jet turbulent à section carrée dans une cavité sont présentés et validés en terme de profils de vitesse moyenne et d'intensité turbulente.

ABSTRACT

In the recent years, the Lattice Boltzmann Method has gained credit as a valuable tool for fluid dynamic simulations. In this context, this thesis aims to verify and validate a developed LBM implementation for several test cases accounting for various flow regimes. The present work is divided in four parts.

At first, a concise literature review provides information about the origin and state of the art of the LBM, with particular emphasis on turbulence modeling. The collision and streaming processes, which constitute the core of fluid dynamics modeling within the LBM, are also described, together with the main related equations.

In the second part, the different steps for the implementation of a LBM code are identified and detailed. Particular emphasis is put on the steps to undertake when switching from the physical world to a lattice-based domain and on the detailed description of the boundary conditions. The Multiple Relaxation Times collision model, able to enhance the LBM stability, is also introduced.

In the third part, results from the developed two-dimensional LBM implementation are presented. Firstly, the Poiseuille flow is simulated and compared to the corresponding analytical solution. Two different boundary conditions are compared in terms of accuracy, and a study on the order of convergence is performed. Then, the Couette-Poiseuille and lid-driven cavity flows are implemented and compared to analytical and experimental results respectively. The unsteady flow past a square obstacle is also presented and validated by comparison to results obtained with the Finite Volume Method.

The fourth and last part is dedicated to three-dimensional simulations. Firstly, the main changes with respect to a two-dimensional implementation are outlined. Then, two pressure driven, steady, three-dimensional flows are simulated and the results compared to analytical solutions. A study on the order of convergence of the three-dimensional LBM is also presented. Later, the challenge of turbulence modeling within the LBM is introduced. The main theoretical aspects of the Large Eddy Simulation turbulence model are outlined. Lastly, the results of a three-dimensional simulation of a turbulent square jet in a cavity are compared to experimental results in terms of average velocity profiles and turbulence intensities.

TABLE OF CONTENTS

DEDICATION	iii
ACKNOWLEDGMENTS	iv
REMERCIEMENTS	v
RÉSUMÉ	vi
ABSTRACT	vii
TABLE OF CONTENTS	viii
LIST OF TABLES	x
LIST OF FIGURES	xi
NOMENCLATURE	xiv
LIST OF APPENDICES	xv
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 LITERATURE REVIEW	2
2.1 Introduction	2
2.2 From The Lattice Gas Automata...	3
2.3 ... to the Lattice Boltzmann Method	4
2.4 LBM Applications	7
2.5 Objectives	8
CHAPTER 3 THE LATTICE BOLTZMANN METHOD	10
3.1 Lattice Characteristics	10
3.2 Structure of a LBM Algorithm	12
3.2.1 Discretization of the physical domain and nondimensionalization of the related parameters	13
3.2.2 Choice of simulation parameters	15
3.2.3 Domain initialization	15
3.2.4 Collision step	16

3.2.5	Streaming step	16
3.2.6	Application of the Boundary Conditions	19
3.2.7	Calculation of the macroscopic parameters and new iteration	25
3.3	The MRT Method	26
3.4	Conclusion on the LBM implementation	29
CHAPTER 4 TWO-DIMENSIONAL LBM VERIFICATION AND VALIDATION .		30
4.1	Introduction	30
4.2	Poiseuille Pressure Driven Channel Flow	33
4.3	Couette-Poiseuille Channel Flow	40
4.4	Lid-Driven Cavity Flow	42
4.5	Unsteady Flow Past Square Obstacle	55
CHAPTER 5 THREE-DIMENSIONAL LBM VERIFICATION AND VALIDATION		68
5.1	Description of Three Dimensional LBM	68
5.1.1	Three-Dimensional Multiple Relaxation Times collision	71
5.2	Pressure Driven Flow	73
5.2.1	Validation 1: Flow between infinite, parallel plates	73
5.2.2	Validation 2: Pressure Driven Channel flow	76
5.3	Square Turbulent Jet	81
5.3.1	Turbulence in the LBM	82
5.3.2	LBM simulation results	89
CHAPTER 6 CONCLUSION AND FUTURE WORKS		101
REFERENCES		103
APPENDICES		107

LIST OF TABLES

Table 4.1	Poiseuille flow: grid characteristics	37
Table 4.2	Poiseuille flow: convergence parameters	38
Table 4.3	Primary and bottom corners' vortex positions at varying Reynolds compared with results from [15]	48
Table 4.4	Primary and bottom corners' vortex positions at $Re=10000$ compared with results from [15]	49
Table 4.5	Lid-driven cavity: grid characteristics	52
Table 4.6	Lid-driven cavity: convergence parameters	52
Table 5.1	Square duct flow: grid characteristics	80
Table 5.2	Square duct flow: convergence parameters	80
Table A.1	Lattice comparisons	108
Table D.1	CPU Time Measurements for the various test cases	113

LIST OF FIGURES

Figure 3.1	The D2Q9 molecule	11
Figure 3.2	Structure of typical LBM algorithm	12
Figure 3.3	Pre-streaming configuration: the highlighted directions are located on the central node	17
Figure 3.4	Post-streaming configuration: the highlighted directions have streamed towards their neighboring nodes	18
Figure 3.5	Pre- and Post-streaming of highlighted distributions at boundary. Missing distributions are dashed	20
Figure 3.6	Steps in halfway bounce-back: pre-streaming, post-streaming and bounce-back implementation. Dashed nodes are fluid, continuous ones are solid. Wall is placed in between nodes	22
Figure 3.7	Missing distributions at Eastern Wall boundary to be calculated with Zou-He method	23
Figure 4.1	Geometry for the Poiseuille Flow	33
Figure 4.2	Streamwise velocity	36
Figure 4.3	Streamwise velocity profile comparison between LBM simulation and analytical solution	36
Figure 4.4	Logarithmic error plot for SRT collision	38
Figure 4.5	Logarithmic error plot for MRT collision	39
Figure 4.6	Geometry for the Couette-Poiseuille Flow	40
Figure 4.7	Velocity for the Couette-Poiseuille flow along vertical direction	41
Figure 4.8	Geometry of the lid driven cavity	43
Figure 4.9	Total Velocity profile at $Re=1000$	45
Figure 4.10	Streamlines at $Re=100$	45
Figure 4.11	Streamlines at $Re=400$	46
Figure 4.12	Streamlines at $Re=1000$	46
Figure 4.13	Streamlines at $Re=10000$	47
Figure 4.14	Comparison between BGK-SRT collision, MRT collision and [15] results for $Re=100$	50
Figure 4.15	Comparison between BGK-SRT collision, MRT collision and [15] results for $Re=400$	50
Figure 4.16	Comparison between BGK-SRT collision, MRT collision and [15] results for $Re=1000$	51

Figure 4.17	Comparison between BGK-SRT collision, MRT collision and [15] results for $Re=10000$	51
Figure 4.18	Comparison between extrapolated velocity profile, velocity on most refined mesh and experimental solution by [15] for horizontal centerline and $u_y(x)$	53
Figure 4.19	Comparison between extrapolated velocity profile, velocity on most refined mesh and experimental solution by [15] for horizontal centerline and $u_x(y)$	54
Figure 4.20	Geometry (not in scale) of flow past a square obstacle	56
Figure 4.21	Streamwise velocity at $Re=10$	57
Figure 4.22	Streamwise velocity at $Re=100$	58
Figure 4.23	Streamlines at $Re=1$	58
Figure 4.24	Streamlines at $Re=10$	59
Figure 4.25	Streamlines at $Re=100$	59
Figure 4.26	Normalized recirculation length vs. Reynolds Number	60
Figure 4.27	Comparison between LBM and FVM for Steady Drag coefficient . . .	62
Figure 4.28	Streamwise velocity along the channel's length	63
Figure 4.29	Cross-stream velocity along the channel's length	63
Figure 4.30	Streamwise velocity at three different locations in the channel's length	64
Figure 4.31	Cross-stream velocity at three different locations in the channel's length	64
Figure 4.32	Lift coefficient evolution over time for $Re=100$	65
Figure 4.33	Drag coefficient evolution over time for $Re=100$	66
Figure 4.34	Drag Coefficient for steady ($Re < 60$) and unsteady ($Re > 60$) flow .	66
Figure 4.35	Strouhal numbers at different Re	67
Figure 5.1	Generic cubic lattice unit. Area 0 cells are black and star-shaped, Area 1 cells are red squares, Area 2 cells are violet triangles and Area 3 cells are blue circles	69
Figure 5.2	D3Q19 cubic lattice from [45]	70
Figure 5.3	Geometry configuration	73
Figure 5.4	Flow between parallel plates: streamwise velocity profile	74
Figure 5.5	Flow between parallel plates: relative and absolute error trends . . .	75
Figure 5.6	Geometry configuration	76
Figure 5.7	Square Duct flow: absolute and relative error trend	77
Figure 5.8	Comparison between quasi-analytical and LBM velocity profile	78
Figure 5.9	Square Duct flow: 3D plot of the streamwise speed	79
Figure 5.10	Turbulence sketch by da Vinci	82

Figure 5.11	LBM simulation of turbulent jet at $Re=184000$. Normalized centerline velocity at a certain position along the channel's length	83
Figure 5.12	Geometry configuration	89
Figure 5.13	Square Turbulent Jet: streamwise normalized velocity at the centerline. BGK-SRT collision operator	92
Figure 5.14	Square Turbulent Jet: streamwise normalized velocity at the centerline. MRT collision operator	92
Figure 5.15	Square Turbulent Jet: averaged streamwise velocity in the domain (MRT collision operator).	93
Figure 5.16	Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 0.28$. BGK-SRT collision operator	94
Figure 5.17	Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 0.28$. MRT collision operator	94
Figure 5.18	Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 1.121$. MRT collision operator	95
Figure 5.19	Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 2.658$. MRT collision operator	95
Figure 5.20	Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 7.088$. MRT collision operator	96
Figure 5.21	Square Turbulent Jet: jet half width	97
Figure 5.22	Half-velocity contour for the $x = 0.003 \cdot h$ and $x = 3.6 \cdot h$ sections . .	98
Figure 5.23	Square Turbulent Jet: Turbulence streamwise intensity at $\frac{x}{D_e} = 0.28$ and $\frac{x}{D_e} = 2.658$	99
Figure 5.24	Square Turbulent Jet: Turbulence spanwise intensity at $\frac{x}{D_e} = 0.28$ and $\frac{x}{D_e} = 2.658$	100
Figure C.1	Geometries obtainable for variable n with unitary aspect ratio ϵ . . .	110

NOMENCLATURE

BGK-SRT	Bhatnagar-Gross-Krook Single Relaxation Time collision operator
CFD	Computational Fluid Dynamics
C_s	Speed of sound in the lattice
DmQn	Lattice m-dimensional with n associated discrete velocities
DNS	Direct Numerical Simulation
\mathbf{e}	Velocity vector
\mathbf{e}_i	Discrete velocity vector
f	Distribution functions
f^{eq}	Equilibrium distribution functions
FVM	Finite Volume Method
GCI	Grid Convergence Index
k_B	Boltzmann constant
LBM	Lattice Boltzmann Method
LES	Large Eddy Simulation
Kn	Knudsen number
LGA	Lattice Gas Automata
Ma	Mach number
MRT	Multiple Relaxation Times collision operator
N-S	Navier-Stokes
RANS	Reynolds Averaged Navier-Stokes
Re	Reynolds number
rms	Root mean square
SGS	Sub-Grid Scales
St	Strouhal number
T	Temperature
\mathbf{x}	Position vector

LIST OF APPENDICES

Appendix A	Lattice Comparisons in 3D	107
Appendix B	MRT matrix for D3Q19 cubic lattice	109
Appendix C	Square Duct Flow: Quasi-Analytical Solution	110
Appendix D	CPU Time Measurements	112

CHAPTER 1 INTRODUCTION

The Lattice Boltzmann method (LBM) is a recent and still developing tool for the simulation of complex fluid dynamic problems. Its main innovation when compared with the other common computational fluid dynamic solvers such as finite volumes, finite elements and finite differences, is that, instead of solving the macroscopic Navier-Stokes equations, it focuses on constructing simplified kinetic models that incorporate the physics of microscopic processes from which the macroscopic flow characteristics are computed. Therefore, the simulation of a limited amount of particles enables a precise fluid behavior prediction in a variety of different situations. The main advantages of the LBM are related to a relatively low implementation difficulty and a particular ease in terms of algorithm parallelisation, which makes it a promising tool for the future of CFD.

The present work is organized as follows. Firstly, a brief and concise literature review will cover the basics of the two dimensional and three dimensional LBM. The main mesh characteristics will be illustrated together with the boundary conditions which can be implemented. Some methods to enhance the stability and performance of the LBM will also be introduced. Then, a few two-dimensional test cases will be described and used for the step-by-step verification of the LBM with a particular focus on the implementation techniques. Then we will shift our attention towards the more innovative and less common three dimensional applications, with a particular attention to the turbulence modeling. In particular, 3D pressure driven flows and a 3D square turbulent jet will be investigated.

CHAPTER 2 LITERATURE REVIEW

2.1 Introduction

Fluid mechanics is a complex and fascinating subject with an unlimited set of different applications. As most advanced engineering problems it does not only involve solutions of equations or numerical calculations, since the correct analysis of a problem is deeper and more complex.

The study of a problem includes the formulation of reasonable hypothesis, the application of the correct laws of physics in the appropriate way and the finding of the related numerical expressions. Then usually two different models have to be defined: the mathematical model and the numerical one. The first one includes several differential equations resulting from theoretical knowledge such as mass, energy or momentum conservation. The most widely employed mathematical model in fluid dynamics is the set of Navier-Stokes equations in which the macroscopic fluid characteristics are expressed in terms of a system of differential equations. What happens at the molecular level is neglected, since no modelling of the more or less organized interactions between the different particles is performed. The numerical model, on the other hand, focuses on the method to use in order to solve the mathematical model, namely on how to approximate the differential equations whether by finite elements, finite volume or finite differences.

In the following section we will explore the origin, premises and theoretical foundations of the Lattice Boltzmann Method, a newly established and developing mathematical model able to quench some of the issues related to the more classical and widespread solutions. A brief outline of the turbulence modeling in fluid dynamics with a focus on turbulence modeling with the LBM will also be provided.

2.2 From The Lattice Gas Automata...

Fluid dynamics problems can be studied using the classical Navier-Stokes model. Another approach is the solution of the Boltzmann Kinetic Equation, based on the principle that the macroscopic dynamics of a fluid are the result of the collective averaged behavior at the molecular level [22]. This procedure, however, requires calculating each and every particle's distribution functions, a long and complex process. This issue can be partially solved by ignoring the molecular level interactions and grouping part of the molecules together. In such a way the kinetic equation can be simplified [21].

The Lattice Gas Automata, developed since the 1970s uses this premise [31, 43]. In the LGA, time, space and particle velocities are discrete [6]. The spatial domain is discretized by means of a regular finite grid of lattice nodes and the time is divided into time steps. Each one of the nodes has a set of judiciously chosen discrete velocity directions, to which are associated cells, each one with a finite number of possible states, such as on and off (occupied or not). We can think of these cells as if occupied by algorithmic entities which are found (or not) on the available cells of the grid and interact with their neighbors. At every time step, these entities will evolve following certain rules, according to their own state and that of their neighbors.

In the LGA the occupation pattern of the cells is described by means of Boolean variables $n_i(x, t)$ (where i indicates the various available directions of the particles)[10]. The various displacement possibilities for the particles depend on the velocity direction associated to their cell: each cell is linked to the corresponding cell of the adjacent nodes by means of the different velocity directions. These depend on the lattice type that is employed, which should in any case be symmetric enough to grant the domain's isotropy, as observed by [31].

However, the steps necessary for the implementation of the LGA present numerous approximations and the LGA is not competitive with respect to the classical Navier-Stokes procedures, due to the lack of precision and stability. In particular the noise in the results and the difficulty encountered for its 3D extension do not make it a viable method for fluid simulations [1].

2.3 ... to the Lattice Boltzmann Method

With respect to the Lattice Gas Automata, the LBM introduces a major improvement: the boolean occupation variables are substituted by distribution functions, a procedure which eliminates statistical noise in the calculations. In fact, it was suggested [28] to replace the Boolean occupation pattern of the LGA with continuous distribution functions, one for every particle, which neglect single particle's motion and correlation. This constitutes the one big change with respect to the LGA and is the foundation of the LBM. The distribution functions will be indicated as:

$$f(\mathbf{x}, \mathbf{e}, t)$$

and give the probability of finding a certain particle at a certain time, t , position \mathbf{x} with a certain speed \mathbf{e} .

In the LBM, the relations between the distribution functions are regulated by the Boltzmann transport equation:

$$\frac{\partial f}{\partial t} + \mathbf{e} \cdot \nabla_{\mathbf{x}} f + \frac{\mathbf{F}}{m} \nabla_{\mathbf{x}} f = \Omega(f) \quad (2.1)$$

The first two terms on the left hand side of Equation 2.1 represent the streaming process and the term \mathbf{F} accounts for the external forces (when present). The term $\Omega(f)$ is the collision operator, which regulates the interactions between particles. Its modeling is crucial since the colliding particles have to be correctly reassigned to their new trajectories. Due to the high number of unknowns present during a collision, it is not easy to fully determine the collision operator. Some approximations have to be made, keeping in mind that its correct interpretation is very important to grant sufficient accuracy. For the LBM, we consider the following hypotheses:

- Only the collisions between two particles are considered.
- Particles are supposed independent before and after the collision. The interactions between particles are therefore neglected, except during the collision.
- The collision process is modeled as instantaneous and uninfluenced by external factors.

These hypotheses suggest that the Boltzmann Transport Equation is only valid for a very diluted fluid, because they imply that the molecules are quite rarefied in the domain. In

reality it was found that the density is not really the crucial parameter to consider. In fact the Knudsen number, Kn , which can be expressed as:

$$Kn = \alpha \frac{Ma}{Re} = \frac{\lambda}{L} \quad (2.2)$$

is the main parameter to consider when judging the applicability of the LBM. The term α is a proportionality parameter linking the ratio between the Mach number and the Reynolds number to the Knudsen number. The term λ indicates the average distance traveled by a random particle between two collisions and L is the characteristic length of the fluid domain. Summarizing, we can say that Kn expresses the mean number of collisions affecting a particle in a certain time and that the collision operator's simplification is only valid for small Kn numbers [27]. This is the case for most incompressible flows, even with turbulence.

A further simplification of the collision operator was suggested by [34] who applied the assumption of a single relaxation time in the collision process, as proposed by [2], to the LBM. In fact the Bhatnagar-Gross-Krook (BGK) collision is currently the most employed operator in the LBM [8, 39]. The BGK model expresses the collision operator as a relaxation of the distribution functions f towards the equilibrium distributions f^{eq} , weighted by the single relaxation time (SRT), τ , which expresses the mean time between two collisions. Concerning the equilibrium distribution functions f^{eq} , they are obtained from the Maxwell distribution [39]:

$$f^{eq} = \frac{\rho}{m} \left(\frac{m}{2\pi k_B T} \right)^{3/2} \exp \left(-\frac{m(\mathbf{e} - \mathbf{u})^2}{2k_B T} \right) \quad (2.3)$$

calculated for null \mathbf{u} speed. This means that the relaxation process is expressed as a relaxation towards an equilibrium state of null speeds. This approximation is clearly valid only for the case of low velocity fluids: the BGK collision is therefore only applicable to incompressible flows.

The collision operator can, then, be written as:

$$\Omega(f) = -\frac{f(\mathbf{x}, \mathbf{e}, t) - f^{eq}(\mathbf{x}, \mathbf{e}, t)}{\tau} \quad (2.4)$$

In the traditional implementation of the LBM-BGK collision operator, relaxation time τ is constant (hence the SRT) and related to the fluid viscosity. The BGK simplified version of the collision operator, even if valid only for incompressible flows, has proven sufficient for the mass, energy and momentum conservation.

To implement a discrete numerical method for the solution of the fluid dynamics, we translate the Boltzmann transport equation from a continuous to a discrete world. In particular we discretize the infinite possible speed directions \mathbf{e} , reducing them to a finite number \mathbf{e}_i , avoiding a too big approximation which would affect the results' accuracy.

The criteria for such an operation are [39]:

- The discretization should be isotropic. The chosen set of velocities should be therefore symmetric.
- The mass, momentum and energy should be conserved despite the discretization

Several types of discretizations which fulfill these requirements exist, the most common being the two-dimensional nine velocity lattice, D2Q9, and the three-dimensional nineteen velocities lattice, D3Q19. These are also the ones which will be employed for the present work. If we consider that the 3D velocity space is now identified by the finite velocities $\mathbf{e}_i = (e_{i,x}, e_{i,y}, e_{i,z})$ we can expand the terms in equation 2.1 as:

$$\begin{cases} \frac{\partial f_i}{\partial t} = \frac{f_i(\mathbf{x}, t + \delta t) - f_i(\mathbf{x}, t)}{\delta t} \\ \mathbf{e}_i \nabla_{\mathbf{x}} f_i = e_{i,\alpha} \frac{\partial f_i}{\partial x_\alpha} = e_{i,\alpha} \frac{f_i(x_\alpha + \delta x_\alpha, t + \delta t) - f_i(x_\alpha, t + \delta t)}{\delta x_\alpha} \end{cases} \quad (2.5)$$

where α represents the Cartesian directions. Then, since the speed is equal to the ratio between the geometric and time steps, $\mathbf{e}_i = \delta \mathbf{x}(\delta t)^{-1}$, we can rewrite the Boltzmann equation with BGK collision (and without the external force \mathbf{F}) in a compact discrete way:

$$f_i(\mathbf{x} + \mathbf{e}_i \delta t, t + \delta t) - f_i(\mathbf{x}, t) = -\frac{\delta t}{\tau} [f_i(\mathbf{x}, \mathbf{e}_i, t) - f_i^{eq}(\mathbf{x}, \mathbf{e}_i, t)] \quad (2.6)$$

\mathbf{x} is the position vector, i represents the different directions compatible with the lattice and \mathbf{e}_i represents the corresponding velocities.

In Equation 2.6, we can identify the footprint of the main processes crucial to the LBM: streaming and collision. In fact, the term on the left side of the equation describes the movement (or stream) occurring in the velocity space of the f_i distributions, which shift from node to neighboring node of the discrete domain according to their velocity \mathbf{e}_i . The right-hand side of the equation can in turn be thought of as a weighting of the streaming to include the variations involved in the collision process. From Equation 2.6, through the

Chapman-Enskog expansion, the N-S equations can be recovered, provided the assumption of negligible density variation (incompressible flow) is maintained [17].

In the same years of the formulation of LBM-BGK a new multiple-relaxation (MRT) lattice Boltzmann equation was developed by [11]. It enabled to overcome some of the LBM-BGK issues, such as fixed Prandtl number and fixed kinematic to bulk viscosity ratio. It was also observed by [24] that the MRT improvement to the original LBM-BGK also increased numerical stability at the cost of a slightly decreased computational efficiency. A detailed description of the MRT method is provided in Section 3.3.

2.4 LBM Applications

Several test cases were used in the literature in order to validate the LBM, keeping in mind that it is a viable option for the simulation of incompressible flows only. Other than the common Poiseuille or Couette flow benchmark cases, [29] used a 2D lattice to simulate the flow in a two dimensional cavity comparing his results with the work by [33]. Accurate cavity flow LBM simulations were also successfully performed by [19], who compared vortex positions with the benchmark calculations of [15]. [19] used the three-dimensional, fifteen velocity LBM lattice [34, 8] to simulate unsteady flow in a three dimensional cavity.

The case of the laminar flow around an obstacle was studied extensively for a circular obstacle [16, 30, 40] using the LBM. Results show reasonable agreement in terms of average drag coefficient and Strouhal number. Analyses of the flow past a square obstacle in a duct are scarcer. LBM simulations of such a flow were performed by [5] and compared to finite volume results, showing good agreement in terms of velocity profiles, mean drag coefficient and Strouhal number.

In general, the LBM presents some positive and encouraging features which make it a candidate for turbulent flow simulations [1]:

- The collision operator is linear. In fact the N-S equations' non linearity is much more time consuming to be solved numerically than the linear collision operator in the LBM.
- In the LBM method, the velocity space is discrete and only a relatively small number of velocity directions are needed in order to recover the macroscopic behavior.
- The pressure determination does not require the solution of a Poisson equation, which is a costly operation.

- The LBM is characterized by a very compact implementation and involves mostly simple operations.
- Because of the nature of both the velocity discretization and the collision step, the LBM can be very easily parallelized.

However, some limitations do exist:

- The applicability of the LBM method is at present limited to incompressible flows meaning that the Mach number should be smaller than 0.3. It can therefore be used for the simulation of incompressible turbulent flows only.
- The basic LBM implementation is limited to uniform, Cartesian lattices and if the lattice were to be made non-uniform, interpolations would be required for the determination of particle distributions on the nodes. This procedure would be costly in terms of computational efficiency and also could cause some loss in accuracy.

The LBM can be used for the direct numerical simulation (DNS) of high Reynolds number fluid flows, since it is capable of altering the viscosity. However due to the time and memory required, DNS is most of the times not a viable solution, even if in recent years it is gaining more and more relevance for specific turbulent flows applications [1].

In the LBM field, large eddy simulations (LES) are popular due to the small overhead in terms of calculation cost required by their implementation [14]. Recently, good results were obtained in simulation of the popular benchmark case which is the case of a surface mounted cube in a channel [23]. Another typical turbulent case studied extensively, under an experimental point of view, is the jet flow [35]. [45] performed a pioneering analysis of the turbulent square jet using the MRT collision model. [14] also investigated the LES turbulence model for simulating the jet flow. They also proposed an alternative approach to the simulation of turbulent flows by relying on the non-physical dissipation of the numerical scheme rather than modelling the smaller scales with an equivalent viscosity, characteristic of the LES.

2.5 Objectives

From the literature review, we can see that the LBM has been used in the recent years for the simulation of several 2D laminar cases, such as the Poiseuille flow, the lid-driven cavity flow and the flow past an obstacle. It would be interesting to investigate more in depth the performance of different no-slip boundary conditions and to explore the capabilities of the MRT collision, with respect to the BGK-SRT collision. We also notice that 3D simulations

with the LBM are scarcer in literature. In particular, an in depth verification and validation of laminar test cases starting from the 2D up to 3D applications is needed. Turbulence modeling with the LBM is in the early stages and not many 3D turbulent simulations exist. The capabilities of the LBM in this area should, then, also be further explored.

The goal of the present work is the development and the rigorous step-by-step verification and validation of a LBM implementation. In particular, we would like to compare results obtained via the LBM with analytical, experimental or other numerical methods' results in a variety of different cases, both 2D and 3D. The objectives are the following:

- Develop a standard 2D LBM code and investigate the performance and effectiveness of methods aiming to improve the original implementation.
- Validate the results with benchmark cases such as the Poiseuille and Couette-Poiseuille flows in a duct and the well referenced lid-driven cavity flow.
- Investigate the applicability of the LBM code for the study of the unsteady flow past a square obstacle.
- Develop a standard 3D LBM code and investigate the performance and effectiveness of methods aiming to improve the original implementation.
- Validate the results with 3D benchmark cases for which analytical solutions exist (3D flow between infinite parallel plates, duct flow).
- Integrate the LES, Smagorinsky based, turbulence model to the 3D LBM code.
- Validate the results by comparison to well-established experimental results.

CHAPTER 3 THE LATTICE BOLTZMANN METHOD

In this chapter, the Lattice Boltzmann method will be discussed in detail, with a particular emphasis on its implementation. We will first begin with a description of the lattice characteristics and then we will shift our attention to the step-by-step implementation. We will also describe the application of physical boundary conditions to the LBM domain and the possible improvements to the original standard implementation. In order to straightforwardly describe the method and avoid unnecessary distractions, the discussion will be focused on the 2D procedure: the 3D and turbulence model extensions will be discussed in Chapter 5.

3.1 Lattice Characteristics

As we have previously stated, the interaction process between molecules is regulated by Equation 2.6, which has to be applied to all of the velocity directions, \mathbf{e}_i . The number of discrete directions, used to reduce the infinite number of possible velocity directions available for each particle in the physical fluid-dynamics, depend on the lattice characteristics, namely on its dimensions (2D or 3D) and on its velocity branches. The most common lattice types are the two dimensional, nine velocity lattice (D2Q9), shown in Figure 3.1 and the three dimensional nineteen velocity lattice (D3Q19), which have also been adopted in this work.

We will at first focus on the two dimensional, nine directional lattice characteristics, since the three dimensional or other lattice's characteristics are very similar. The D2Q9 lattice allows nine different velocity directions, including the four orthogonal Cartesian speeds (positive and negative), the four associated to the two bisectors (positive and negative) and a state of rest. It is useful, for a clearer discussion of the implementation, to label these speeds with numbers from 1-9, in the order most common in literature, starting from the center, as shown in Figure 3.1.

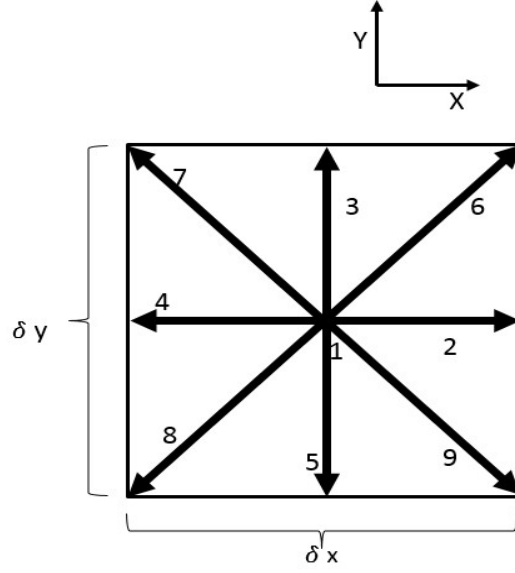


Figure 3.1 The D2Q9 molecule

The associated velocity magnitudes in terms of the Cartesian axes (x, y) are:

$$\begin{cases} e_x = [0, 1, 0, -1, 0, 1, -1, -1, 1] & \text{for } i = 1, \dots, 9 \\ e_y = [0, 0, 1, 0, -1, 1, 1, -1, -1] & \text{for } i = 1, \dots, 9 \end{cases} \quad (3.1)$$

In the LBM, space is discretized as a regular lattice ($\delta x = \delta y$), and the most common approach in literature is using a non-dimensional domain. This implies that if $c = \delta x / \delta t = 1$, then $\delta x = \delta t$. In non-dimensional form $\delta x = \delta t = 1$, which allows for great simplification of the formulas. In practice, this means that the neighboring nodes of a generic node \mathbf{x} are located at a distance of $\mathbf{x} + \mathbf{e}_i$. As a consequence, the particles located on the orthogonal direction will have a speed of 1 lattice units/time step while the ones located on the bisectors will have a velocity magnitude of $\sqrt{2}$ lattice units/time step.

The nodal fluid density and momentum are obtained as the sum of the directional distributions:

$$\rho = \sum_{i=1}^9 f_i \quad \rho \mathbf{u} = \sum_{i=1}^9 f_i \mathbf{e}_i \quad (3.2)$$

The nodal velocity is in turn obtained as the sum of directional velocities weighted by the

directional distributions, divided by the local density:

$$\mathbf{u} = \frac{1}{\rho} \sum_{i=1}^9 f_i \mathbf{e}_i \quad (3.3)$$

3.2 Structure of a LBM Algorithm

In order to guide the reader into the different steps of a standard LBM implementation, Figure 3.2 provides a concise flow chart scheme whose various steps will be discussed in detail in this section.

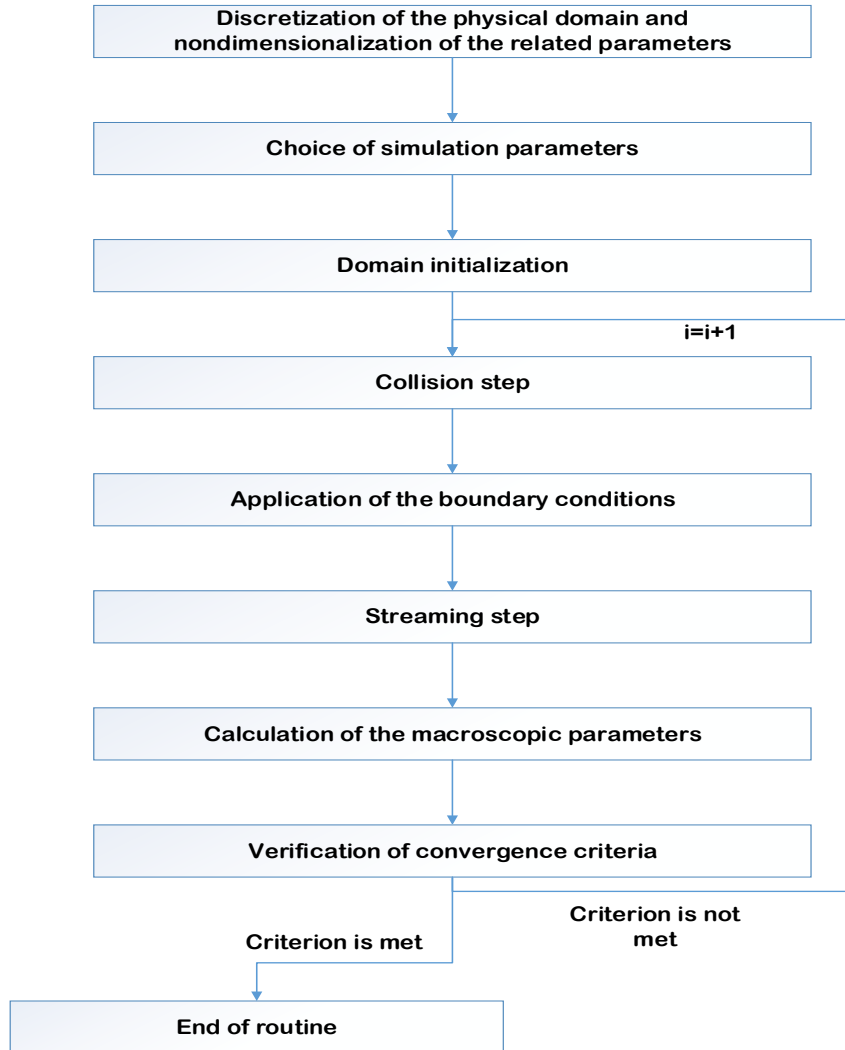


Figure 3.2 Structure of typical LBM algorithm

3.2.1 Discretization of the physical domain and nondimensionalization of the related parameters

Lattice Boltzmann simulations should correctly model reality and so they should be able to correctly reflect the physics of an actual system. This can sometimes prove challenging since all LBM simulations are performed with lattice variables and lattice units. In particular we have two constraints to respect when choosing the units for our simulations:

- The simulated system should be geometrically similar to the physical one
- The chosen parameters should allow to reach an acceptable accuracy

The switching from the physical system to the lattice one is performed in two steps:

1. The physical system is changed into a dimensionless one. For this we need a characteristic length (l_0) and time scale (t_0)
2. The non-dimensional system is translated into lattice units through the choice of discrete time and length steps.

We must keep in mind that, through every step, the Reynolds number should be conserved, in order for us to be simulating the same flow type. In the LBM, the most common approach is to set the lattice units in term of time and length as unitary. As mentioned earlier, this simplifies all the formulas involved. It should be noted that the various density distributions and relaxation times are already expressed in terms of lattice units.

The choice of the characteristic length and time scales for the non-dimensionalization of the physical system are dependent on the problem studied: as an example, for a 2D flow around a cylinder, l_0 and t_0 could be chosen as, respectively, the size of the immersed obstacle and the time needed for the fluid to cover such a distance. For instance, if we consider the time $t_{physical}$ and the position $x_{physical}$, their expressions in the non-dimensional world are:

$$t_{nondimensional} = \frac{t_{physical}}{t_0} \qquad x_{nondimensional} = \frac{x_{physical}}{l_0} \qquad (3.4)$$

The velocity in the physical world can be related to the one of the non-dimensional one by:

$$u_{phy} = \frac{x_{phy}}{t_{phy}} = \frac{x_{nd} \cdot l_0}{t_{nd} \cdot t_0} = \frac{l_0}{t_0} u_{nd} \qquad (3.5)$$

The nondimensional world is characterized by having a unitary reference length $l_{0,nd}$ and reference time $t_{0,nd}$. The Reynolds number has to be unchanged by the transformation from the physical to the non-dimensional world:

$$Re = \frac{x_{phy} \cdot u_{phy}}{\nu_{phy}} = \frac{l_{0,nd} \cdot u_{0,nd}}{\nu_{nd}} = \frac{1}{\nu_{nd}} \quad (3.6)$$

Once the physical system has been made dimensionless, we can decide the relation between the non-dimensional world and the lattice units. The discrete space step Δx is described as the reference length divided by the number of units used for the discretization. If the domain has a size of N lattice units for instance, the distance between two nodes in non-dimensional coordinates would be $\Delta x = 1/N$. The same procedure can be applied for the time step, using the total number of iterations N_{iter} and defining $\Delta t = 1/N_{iter}$.

The speed in lattice units, u_{lb} , can be expressed as:

$$u_{lb} = u_{nd} \frac{N}{N_{iter}} = u_{nd} \frac{\Delta t}{\Delta x} \quad (3.7)$$

and the Reynolds number:

$$Re = \frac{N \cdot N/N_{iter}}{\nu_{lb}} = \frac{\Delta t}{(\Delta x)^2 \nu_{lb}} = \frac{1}{\nu_{nd}} \quad (3.8)$$

From this relation, the viscosity in the lattice can be related to the non-dimensional one by:

$$\nu_{lb} = \nu_{nd} \Delta t / (\Delta x)^2 \quad (3.9)$$

The density can be expressed in lattice units as unitary via the normalization of the physical density by itself. This means that the initial density in the lattice will be unitary and that only small variations around the initial value are expected, given the incompressibility assumption of the method.

Another way to make a reliable comparison between LBM results and physical results is to render both of them nondimensional. Physical results will therefore be translated to the dimensional world and the same will be done with the LBM ones. This approach will often be employed for the present work for its inherent simplicity.

3.2.2 Choice of simulation parameters

In this step, some of the simulation parameters are chosen, such as the grid size, the Reynolds number, the density and viscosity, the relaxation time, the running time or the convergence criterion. In the LBM, the density is most commonly imposed as unitary. The Reynolds number is chosen according to the physical case we want to model. The characteristic velocity of the lattice is set to be well below the compressibility limit, since the LBM is only valid for incompressible flows. The compressibility limit is related to the Mach number defined as the ratio between the maximum speed of the flow in lattice and the speed of sound in the lattice. The speed of sound (C_s) is a lattice constant and equal to the density variation over the pressure variation [25]:

$$C_s^2 = \frac{\delta p}{\delta \rho} \quad \rightarrow \quad C_s = \frac{1}{\sqrt{3}} \quad (3.10)$$

The relaxation coefficient, τ , within the SRT model is unique and applied to all the directions. It is related to the fluid's kinematic viscosity ν [17]:

$$\tau = \frac{\nu}{C_s^2} + 0.5 = 3\nu + 0.5 \quad (3.11)$$

The kinematic viscosity is set according to which Reynolds number we aim to model. Its variation depends solely on the temperature and so, since we are modeling isothermal flows, it is constant.

The convergence criterion is case specific and will be defined according to the type of problem being studied.

3.2.3 Domain initialization

The initial conditions for a typical CFD simulation of incompressible flows are usually in terms of the velocity components, density and pressure. In the LBM, however, these conditions are not sufficient since a certain value must be assigned to the distribution functions. One typical approach to solve this problem is to set the local initial distribution functions as equal to the corresponding equilibrium distributions. These can in fact be calculated using Equation 2.3 given velocity and density.

Just like any CFD simulation, we should then allow a transitory period necessary for reaching the steady state. This way of implementing the initial conditions will be followed for all of

the test cases which will be presented in the next chapters.

3.2.4 Collision step

The post-collision state is calculated via Equation 2.6. Collision of fluid particles is considered as a relaxation towards a local equilibrium, with the equilibrium distribution functions at the equilibrium for the i – th direction defined as [8]:

$$f_i^{eq} = w_i \cdot \rho \left[1 + \frac{\mathbf{e}_i \mathbf{u}}{C_s^2} + \frac{(\mathbf{e}_i \mathbf{u})^2}{2C_s^4} - \frac{\mathbf{u}^2}{2C_s^2} \right] \quad (3.12)$$

where the weights w_a (related to the D2Q9 molecule) are 4/9 for particles at rest, 1/9 for orthogonal directions and 1/36 for the oblique ones. The sum of these weights is obviously equal to one, meaning that the maximum density which can be reached at a certain node is unitary.

3.2.5 Streaming step

In the implementation scheme of the LBM, the streaming step occurs after the boundary conditions are applied. However, for sake of clarity, it will be presented before the boundary conditions.

In the streaming process, the distribution functions resulting from the collision process are streamed to their neighboring nodes according to their directions. If the lattice is uniform and the time step is set in order to achieve a lattice speed of 1, the distributions will travel exactly to their neighboring nodes at every time step.

This process can be visualized in Figures 3.3 and 3.4. In particular, Figure 3.3 shows the pre-streaming configuration, in which the highlighted distributions are located on the central node. Then, these stream to their neighboring nodes, according to their associated velocity direction, and the post-streaming configuration shown in Figure 3.4 is reached.

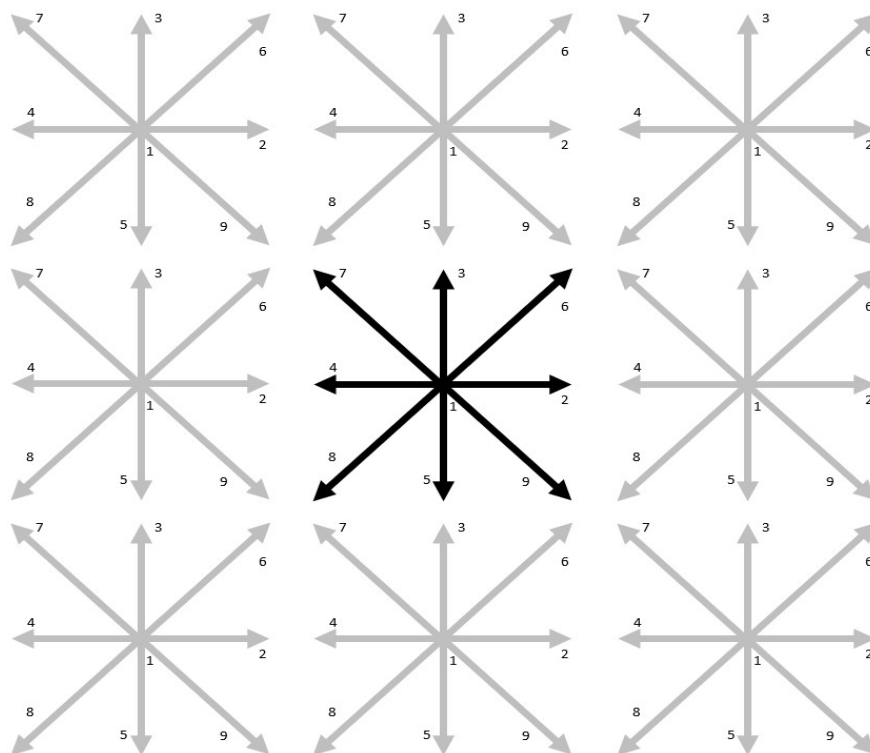


Figure 3.3 Pre-streaming configuration: the highlighted directions are located on the central node

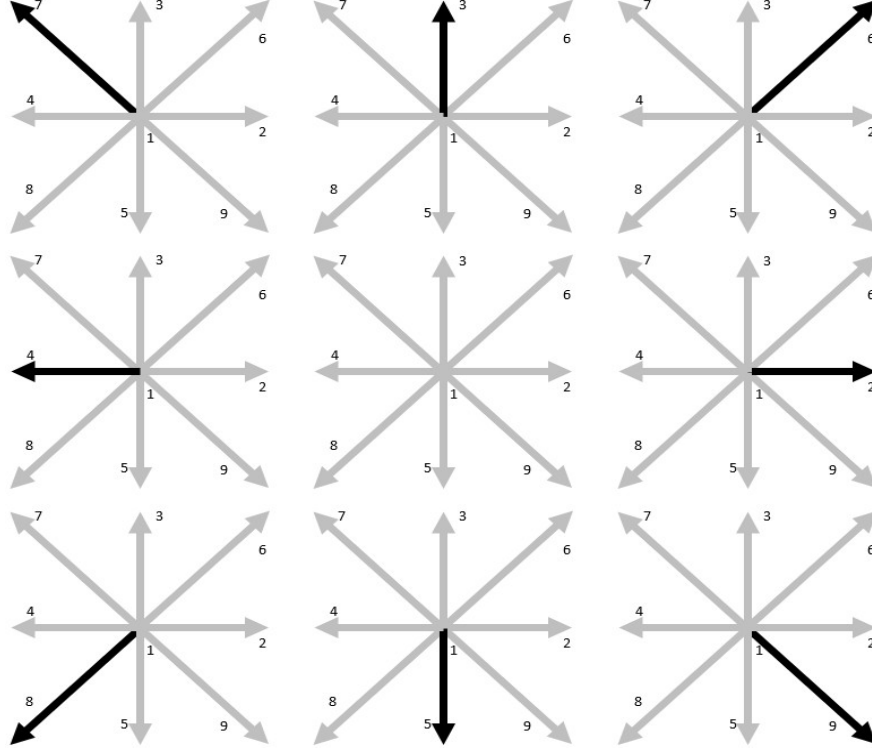


Figure 3.4 Post-streaming configuration: the highlighted directions have streamed towards their neighboring nodes

Applying Equation 2.6 at time t , we find the distribution functions at time $t + 1$ for the new location $\mathbf{x} + \mathbf{e}_i$. In fact, in Equation 2.6, the streaming and collision processes are joined together in one equation only. However, from an implementation point of view, they are split into two separate processes. This means that, when we implement the method, we will first calculate the distributions at time $t + 1$ for the original positions \mathbf{x} and successively stream them towards the new time step $(t + 1)$ positions $\mathbf{x} + \mathbf{e}_i$.

3.2.6 Application of the Boundary Conditions

Right after the collision step and just before the streaming process, the boundary conditions are applied.

The application of the boundary conditions depends on the type of boundary and on the macroscopic property (velocity or pressure) we want to impose. They are chosen in order to correctly model the physical boundaries of the problem and the main challenge consists in translating the macroscopic information to the microscopic distribution functions without affecting the numerical accuracy or stability. We should consider that, when the streaming step of the simulation occurs, the nodes found close to the boundary lack some of their neighbors. Therefore some of the distribution functions which are supposed to stream towards the boundary nodes will be missing. This concept is better explained visually in Figure 3.5. On its left side the pre-streaming configuration is shown, with the highlighted distributions found on the neighbors of the central node. The right-hand side of Figure 3.5 shows the post-streaming configuration, in which the highlighted distributions have streamed on the central node. However, since the eastern neighbors are not present because of the wall, the "dashed" distributions are missing.

Since, in order to evaluate the macroscopic velocity, we need a complete set of distribution functions, we must find a way to calculate the missing ones. Several efficient boundary conditions exist, but in what follows only those which were actually used in our test cases will be presented.

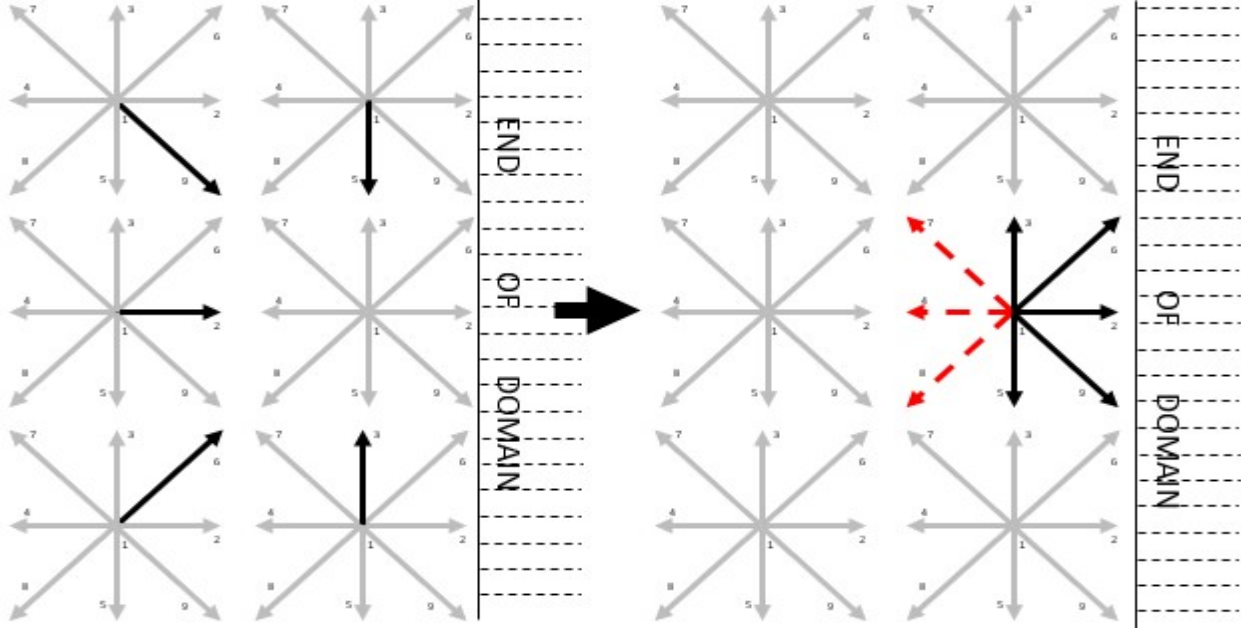


Figure 3.5 Pre- and Post-streaming of highlighted distributions at boundary. Missing distributions are dashed

Periodic boundary conditions

Among all of the possible boundary conditions, the simplest one is certainly the periodic boundary condition, in which the edge nodes behave as if they were attached to the opposite edge nodes. This condition is typically used for modeling steady state domains, in which, since the solution does not vary in time, we can consider that the fluid exiting the domain re-enters from the opposite side. In the Poiseuille flow, as an example, by the use of this boundary condition, we can infer the steady state solution using only very few nodes in the flow direction. Since the boundary nodes are part of the fluid, the periodic boundary is a "wet node" condition. It should be noted that the implementation of the periodic boundary conditions is already accounted for by the streaming step and consequently no additions to the original implementation are required.

Bounce-back boundary conditions

The bounce-back boundary conditions are used to simulate the presence of solid walls. The wall boundary is placed on solid cells just outside the fluid cells belonging to the actual boundary of the flow. The implemented boundary is therefore external to the domain representing the fluid: it then is a "dry node" boundary condition. The bounce-back is implemented

by copying the known distributions of the boundary fluid cells to their unknown neighbors pointing in the opposite direction. It can be "full-way", meaning that all of the distributions are replaced with the ones pointing in the opposite direction or "half-way" where only the unknown distributions are reflected. Both are easy to implement but the "full-way" bounce-back was chosen for the rest of this work due to the fact that it is independent of the wall orientation and so its implementation is the same regardless of the wall's position.

For the D2Q9 lattice it is useful to identify the original direction and the associated opposite directions to which the distributions are reflected. These are:

$$\begin{aligned} \text{directions} &= [1, 2, 3, 4, 5, 6, 7, 8, 9] \\ \text{opposite}_{\text{directions}} &= [1, 4, 5, 2, 3, 8, 9, 6, 7] \end{aligned}$$

and therefore the bounce-back is implemented with the following equation:

$$f_{\text{opposite}}(\mathbf{x}, t) = f_{\text{directions}}(\mathbf{x}, t) \quad (3.13)$$

Figure 3.6 illustrates the three stages of the half-way bounce-back implementation (presented instead of the full-way because it can be more easily visualized graphically):

- Step 1. Pre-streaming configuration: the highlighted distributions are found on the easternmost node of the domain
- Step 2. Post-streaming configuration: the highlighted distributions have streamed on the solid wall nodes.
- Step 3. Bounce-back boundary implementation: the highlighted distributions are reflected back to the original node and occupy directions opposite to the original ones.

One of the reasons why the LBM is promising in simulating flows in complex geometries, such as porous media, is the straightforwardness in the implementation of the bounce-back boundary conditions. Since the wall is not a part of the domain but is placed on nodes found just outside of it, the bounce-back scheme was first thought of as of first-order numerical accuracy, while the Lattice Boltzmann Method for nodes interior to the domain is of second order accuracy. This accuracy issue was solved by [46] who suggested placing the wall in between nodes: this means that the physical wall will be shifted half a mesh unit into the fluid [26] and that the LBM remains second order accurate. In the present work however, for sake of simplicity the classical bounce-back implementation is maintained meaning that the test cases which require a bounce-back condition will have first order accuracy.

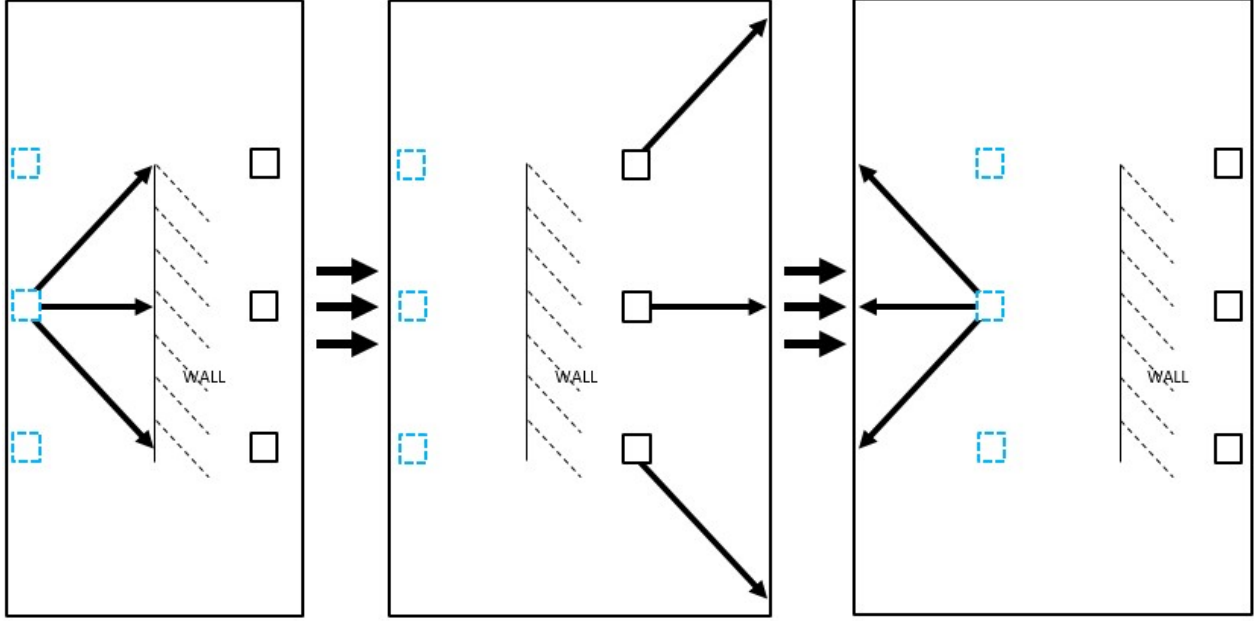


Figure 3.6 Steps in halfway bounce-back: pre-streaming, post-streaming and bounce-back implementation. Dashed nodes are fluid, continuous ones are solid. Wall is placed in between nodes

Zou-He boundary conditions

The Zou-He boundary conditions [47] constrain the flow at boundaries in terms of velocity: this means that they are particularly useful if the objective is the representation of a boundary moving with a given speed. They are found to be second order accurate and place the wall right on the nodes of the domain's boundary [41].

Their description is quite elaborate, even if their implementation is quite straightforward. First of all it should be noted that only the missing distributions on the boundary nodes will be calculated and the remaining ones will be left intact. Also, the boundary nodes are "wet nodes" since they can be thought as if representing the last physical fluid particle adjacent to the wall and so are "fluid". The wall can move both horizontally and vertically and its movement is described by the velocity vector: $\mathbf{u}_0 = \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}$. If we consider a classic Cartesian reference frame and if the boundary is, say, a vertical plate found at the Eastern boundary of the domain, the only distributions subject to change are the ones aiming West. This is due to the fact that if we imagine a streaming process the $f_{4,7,8}(East, t + 1)$ will come from time step t and from the "wall" where no distributions are defined since it does not belong to the domain. The f_4, f_7, f_8 distributions are then the three unknowns we have to solve. For

sake of clarity, we will neglect to indicate the position and time to where they belong since it is understood that it is $f_{4,7,8}(East, t + 1)$. The missing distributions are presented in Figure 3.7

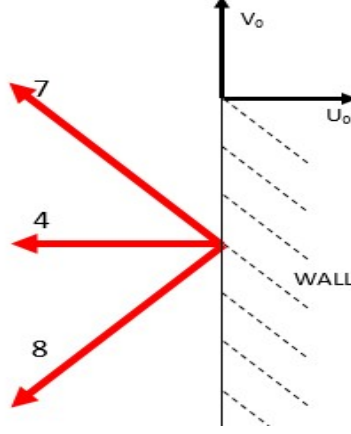


Figure 3.7 Missing distributions at Eastern Wall boundary to be calculated with Zou-He method

We do have several equations we can use to solve for these unknowns: first of all we know that ρ is the sum of all the distribution functions. Also the speed is of course computed as the sum of the distributions, taking into account their different directions. We can therefore write the horizontal and vertical velocity equilibria, together with the density calculation:

$$\begin{cases} \rho = \sum_{i=1}^9 f_i = f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7 + f_8 + f_9 \\ \rho u_0 = f_2 + f_6 + f_9 - f_4 - f_7 - f_8 \\ \rho v_0 = f_3 + f_6 + f_7 - f_5 - f_8 - f_9 \end{cases} \quad (3.14)$$

Zou and He suggested a fourth equation which is valid assuming the bounce-back condition holds in the horizontal direction:

$$f_2 - f_2^{eq} = f_4 - f_4^{eq} \quad (3.15)$$

Going back to the way the equilibrium distributions are computed (Equation 3.12) we find that:

$$\begin{cases} f_2^{eq} &= \frac{1}{9} \cdot \rho \left[1 + 3 \cdot 1 \cdot u_0 + 9/2(1 \cdot u_0)^2 - 3/2(u_0^2 + v_0^2)^2 \right] \\ f_4^{eq} &= \frac{1}{9} \cdot \rho \left[1 + 3 \cdot (-1) \cdot u_0 + 9/2(-1 \cdot u_0)^2 - 3/2(u_0^2 + v_0^2)^2 \right] \end{cases} \quad (3.16)$$

Rewriting the horizontal bounce-back condition we get that;

$$f_2 - f_4 = f_2^{eq} - f_4^{eq} \quad \rightarrow \quad f_2 - f_4 = \frac{1}{9} \cdot \rho (6u_0) \quad \rightarrow \quad f_4 = f_2 - \frac{2}{3}\rho u_0 \quad (3.17)$$

One of the unknowns is now solved. If we put together the horizontal equilibrium and density calculation equation, we obtain an expression of the density dependent only on known distributions:

$$\rho = \frac{f_1 + f_3 + f_5 + 2 \cdot (f_2 + f_6 + f_9)}{1 + u_0} \quad (3.18)$$

We can now solve the vertical equilibrium and finally we get the following expression for the remaining missing distributions:

$$\begin{cases} f_7 = f_9 + \frac{f_5 - f_3}{2} + \frac{\rho v_0}{2} - \frac{1}{6}\rho u_0 \\ f_8 = f_6 + \frac{f_3 - f_5}{2} - \frac{\rho v_0}{2} - \frac{1}{6}\rho u_0 \end{cases} \quad (3.19)$$

The missing distributions have now been related to the existing ones and to the velocity of the wall. In this case, the Zou-He boundary condition was applied to a wall located on the Eastern side of the domain but the same procedure can be applied for boundaries located in any other positions with respect to the flow.

Since, in the incompressible LBM, there exists a precise relation between pressure and density:

$$p = C_s^2 \rho = \frac{\rho}{3} \quad (3.20)$$

we can use a procedure similar to the Zou-He for moving walls in order to calculate the missing distributions when a pressure boundary is applied. In that case, the density, related to the pressure, would be known and so we could reverse Equation 3.18 to obtain the horizontal speed at the boundary and then use the horizontal and vertical equilibria in order to work out the remaining missing distributions.

Equilibrium based boundary conditions

In this case, the unknown distributions are replaced with equilibrium distributions calculated with a density and speed chosen according to the macroscopic properties. An easy way to visualize this is thinking that the existence of a fictional flow with the desired characteristics is imposed just outside of the boundary, and its distribution functions stream inside the domain. It should be noted that, although simple to implement, some error in terms of speed does exist at the boundary [26].

Free Boundary conditions

These conditions are used when we need to implement a fully developed flow boundary condition. In fact when the flow is fully developed, we can say that the speed gradient towards the boundary is null. In terms of LBM this translates to imposing that the distributions on the nodes at the boundary \mathbf{x}_b are equal to the distributions on the nodes adjacent to the boundaries, \mathbf{x}_{b-1} . The associated equation is:

$$f_i(\mathbf{x}_b) = f_i(\mathbf{x}_{b-1}) \quad (3.21)$$

Other boundary conditions

Several other boundary conditions do exist, such as the ones accounting for a curved boundary [4] and others which extrapolate the distributions according to the neighboring nodes [26]. Generally more costly in terms of calculations to implement, in some specific cases they do offer useful results. However, since they will not be used for the remainder of the work, they will not be presented.

3.2.7 Calculation of the macroscopic parameters and new iteration

In this step the macroscopic parameters, typically velocity, density and pressure are calculated using Equations 3.2 and 3.3. Then, if the convergence criterion or the maximum imposed simulation step are not reached, a new iteration is performed.

3.3 The MRT Method

The previous description is referred to the "classical" LBM implementation, with the BGK collision operator and a single relaxation time. However the Multiple Relaxation Times lattice Boltzmann equation has been found to generally result in a greater stability compared to the SRT collision. In fact the MRT implementation also overcomes some issues of the SRT based LBM such as fixed Prandtl number and fixed ratio between kinematic and bulk viscosity [12].

All of this is made possible via the expression of the collision operator in the moment space and the fact that every moment has its own relaxation coefficient, which can be tuned to achieve optimal stability.

The collision equation for the MRT method becomes:

$$f(\mathbf{x} + \mathbf{e}_i \delta t, t + \delta t) - f(\mathbf{x}, t) = -M^{-1} \hat{S} [m(\mathbf{x}, t) - m^{eq}(\mathbf{x}, t)] \quad (3.22)$$

What happens is that the collision process will occur in terms of moment rather than speed, and therefore the velocity based distribution functions must be translated to the moment space. This is achieved by the matrix M , which has, for the D2Q9 case, nine rows and nine columns. This transformation matrix is constructed via the Gram-Schmidt procedure [3]. Its expression is given below:

$$\left\{ \begin{array}{l} M = (\varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5, \varphi_6, \varphi_7, \varphi_8, \varphi_9) \\ \varphi_1 = (e_x^2 + e_y^2)^0 \\ \varphi_2 = -4 + 3(e_x^2 + e_y^2) \\ \varphi_3 = 4 - 6(e_x^2 + e_y^2) + 9(e_x^2 \cdot e_y^2) \\ \varphi_4 = \rho e_x \\ \varphi_5 = 3(e_x \cdot e_y^2) - 2e_x \\ \varphi_6 = \rho e_y \\ \varphi_7 = 3(e_x^2 \cdot e_y) - 2e_y \\ \varphi_8 = e_x^2 - e_y^2 \\ \varphi_9 = e_x e_y \end{array} \right. \quad (3.23)$$

The matrix m representing the distribution functions translated to moment space is obtained via the multiplication:

$$m = f \cdot M \quad (3.24)$$

The collision matrix \hat{S} , is a diagonal matrix containing the relaxation coefficients associated to each of the moments. There is no analytical way to calculate these coefficients: a trial and error analysis based on the LBM-MRT method results must be done. This procedure, however, was not entirely performed in the present work and the most commonly used values of $s_1 = 1$, $s_2 = s_3 = 1.4$, $s_4 = s_6 = 1$, $s_5 = s_7 = 1.2$ found in literature were at first used [12]. The last two coefficients, s_8 and s_9 are obtained from the viscosity with the following equation:

$$s_8 = s_9 = \frac{2}{1 + 6\nu} \quad (3.25)$$

From these values, a reduced linear analysis was performed using as a benchmark the pressure driven Poiseuille flow, since a precise analytical result exists for this particular case. No appreciable result improvement was observed for small variations of the relaxation coefficients with respect to their literature values, and a larger change in these values quickly lead to instability. The values found in literature were, then, judged acceptable.

The relaxation times matrix has then the form:

$$\hat{S} = \text{diag}(s_1, s_2, s_2, s_4, s_5, s_4, s_5, s_8, s_8) \quad (3.26)$$

Another necessary step for the MRT collision operator is the calculation in the moment space of the nine equilibrium distribution functions. These are related to the fluid density, the energy (e) and energy squared (ϵ), the two dimensional moment components (j_x, j_y), the energy fluxes (q_x, q_y) and finally the normal and diagonal strain rate tensors (p_{xx}, p_{yy}). The only hydrodynamic moments which are locally conserved in the velocity to moment space transformation (in general $m_\beta = m_\gamma$) are the scalar density ρ and moment j , since all the others are kinetic non conserved moments ($m_\beta \neq m_\gamma$)

$$\left\{ \begin{array}{l} m_1^{eq} = \rho \\ m_2^{eq} = e = -2\rho + 3(j_x^2 + j_y^2) \\ m_3^{eq} = \epsilon = \rho - 3(j_x^2 + j_y^2) \\ m_4^{eq} = j_x \\ m_5^{eq} = q_x = -j_x \\ m_6^{eq} = j_y \\ m_7^{eq} = q_y - j_y \\ m_8^{eq} = p_{xx} = (j_x^2 - j_y^2) \\ m_9^{eq} = p_{xy} = j_x \cdot j_y \end{array} \right. \quad (3.27)$$

Where j_x and j_y represent the momentum in the x and y directions.

3.4 Conclusion on the LBM implementation

From a numerical point of view, the LBM has several advantages with respect to the classical numerical methods applied to fluid mechanics. Firstly, despite an apparent difficulty in the correct simulation of all of the microscopic interactions in the fluid, the actual method is actually quite simple to implement. This is because of the performed simplifications of the microscopic scale, such as the discretisation of both the domain and the available velocities of every cell, and of the collision process. Most of the boundary conditions are also easily integrated in the LBM, which enables the simulation of the flow in complex geometries. The fact that the various cells have a "local" behavior, since they only interact with the neighboring cells, means that the LBM is ideal for parallelisation. This implies that both the computational capacity and precision of the calculations can be greatly increased. It should also be noted, that since the solution is "time dependent", the LBM can simulate both transient and steady state, stationary or unsteady solutions. However, some disadvantages are also present, mainly linked to the relative novelty of the LBM and the on going research. As an example, compressible flows can't be simulated at present and high Reynolds applications which are characterized by a low viscosity require some special treatments.

CHAPTER 4 TWO-DIMENSIONAL LBM VERIFICATION AND VALIDATION

4.1 Introduction

In this chapter we analyze in depth how the implemented LBM code performs in several 2D cases which include stationary and moving boundaries, pressure gradients, unsteady and steady flows. Different Reynolds numbers and geometries will be done. We will start with the well-known Poiseuille and Couette-Poiseuille flows. Then we will simulate a more complex flow, such as the lid-driven cavity flow, and finally the unsteady stream past a square obstacle will be implemented. All the details about the data of the simulations which were performed will be accurately indicated for them to be fully replicable.

First of all, however, some considerations have to be made, in order for the verification and validation procedure to be rigorous and precise. In fact, when we want to implement a fluid dynamics simulation, we want to make sure that the solution we offer is both convergent and accurate. Several ways exist to measure both these parameters: for instance to measure convergence we could compare two successive iterations of the solution and check whether the difference between the two is close to zero. To verify accuracy we could compare the solution to the analytical one, to solutions obtained via CFD commercial software or to reliable experimental values.

The Navier-Stokes equations have an analytical solution only for few cases, which include the Poiseuille and Couette-Poiseuille flows. We will then calculate LBM simulation's results in these two cases and compare them to the differential equations' solution. For the lid-driven cavity flow the results will be compared to the benchmark reference which is the work of [15]. The unsteady flow past a square obstacle will in turn be compared to the accurate CFD Finite Volume Method computations performed by [5].

One more thing we have to consider when dealing with a fluid dynamics problem, is that it is very important to look at parameters other than the results' comparison in order to ensure that the used method is scientific and reliable. Although no formal procedure is universally recognized by the fluid dynamics community, [36] indicates that the main ones include the method's order of accuracy estimation, the grid independence or grid convergence evaluation and the estimation of the grid-related error.

The work of [36] details a procedure for their estimation, divided into several steps:

- Step 1. Definition of a reference mesh or grid size h . For the one, two and three dimensional cases, its expressions are:

$$\begin{cases} h_{1D} = \left[\frac{1}{N} \sum_{i=1}^N (\Delta L_i) \right]^1 \\ h_{2D} = \left[\frac{1}{N} \sum_{i=1}^N (\Delta A_i) \right]^{1/2} \\ h_{3D} = \left[\frac{1}{N} \sum_{i=1}^N (\Delta V_i) \right]^{1/3} \end{cases} \quad (4.1)$$

where ΔL_i , ΔA_i and ΔV_i are the local single cell's length, area and volume and N is the total number of cells.

- Step 2. Selection of three significantly different grids, named coarse, intermediate and fine. The grid refinement factor, defined as the ratio between the "coarser" and "finer" grid size, should be greater than 1.3 (value based on experience)
- Step 3. Letting $h_1 < h_2 < h_3$ and $r_{21} = \frac{h_2}{h_1}$, $r_{23} = \frac{h_3}{h_2}$, the extrapolated values (ϕ_{ext}^{21} , based on the two finer meshes) and approximate relative errors (e_a^{21} difference between intermediate and fine mesh, e_{ext}^{21} difference between fine and extrapolated mesh) can be

calculated with the following expressions:

$$\left\{ \begin{array}{l} p = \frac{1}{\ln(r_{21})} \left| \ln \left| \frac{\epsilon_{32}}{\epsilon_{21}} \right| + q(p) \right| \\ q(p) = \ln \left(\frac{r_{21}^p - s}{r_{32}^p - s} \right) \\ s = 1 \cdot \text{sign}(\epsilon_{32}/\epsilon_{21}) \\ \phi_{ext}^{21} = \frac{r_{21}^p \phi_1 - \phi_2}{r_{21}^p - 1} \\ e_a^{21} = \left| \frac{\phi_1^{21} - \phi_2}{\phi_1^{21}} \right| \\ e_{ext}^{21} = \left| \frac{\phi_{ext}^{21} - \phi_1}{\phi_{ext}^{21}} \right| \end{array} \right. \quad (4.2)$$

where ϕ is a key variable for the study in object associated to the three meshes. The terms ϵ_{32} and ϵ_{21} are respectively $\phi_3 - \phi_2$ and $\phi_2 - \phi_1$.

- Step 4. Calculate the fine grid convergence index with

$$GCI_{fine}^{21} = \frac{1.25 e_a^{21}}{r_{21}^p - 1} \quad (4.3)$$

This parameter is obtained with the Richardson extrapolation and is used to determine the precision of the solution on the chosen mesh. It gives an indication of the difference between the extrapolated results (associated to an infinitely fine mesh) and the obtained results. Small values of the GCI index indicate that the obtained result is within the range of the asymptotic value.

4.2 Poiseuille Pressure Driven Channel Flow

The first test case is very basic. We want to check the LBM code's behavior when simulating a 2D laminar channel flow driven by a pressure gradient. Details of the geometry in study are provided in Figure 4.1.

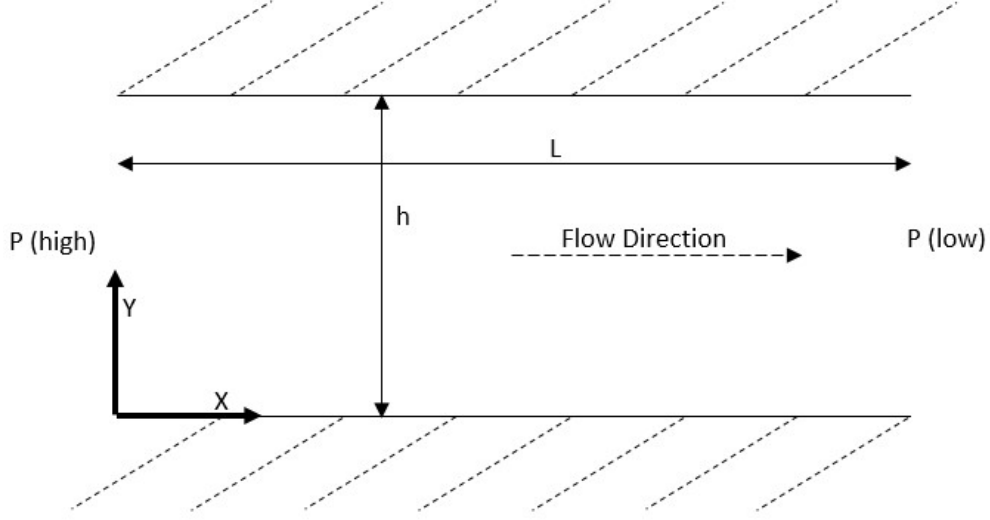


Figure 4.1 Geometry for the Poiseuille Flow

Steady, laminar, incompressible pressure driven fluid motion between two infinite flat, still plates, separated by the distance h , is called Poiseuille flow, from the name of the French physicist Jean Poiseuille. Because of the straightforward nature of such flow, an analytical solution to the N-S equations can be found, given the following boundary conditions:

$$\begin{cases} \text{at } y = 0 & \rightarrow & u_x = u_y = 0 \\ \text{at } y = h & \rightarrow & u_x = u_y = 0 \end{cases} \quad (4.4)$$

The final solution for the velocity profile is:

$$u_x(y) = -\frac{1}{2\mu} \frac{dp}{dx} (hy - y^2) \quad (4.5)$$

where μ is the dynamic viscosity related to the kinematic one by:

$$\mu = \nu \cdot \rho \quad (4.6)$$

If we look at the first order derivative of the velocity profile, we observe the presence of a maximum in correspondence of $y = h/2$.

If we integrate the expression for the velocity u_x from 0 to h and divide the results by the channel's height we obtain the average velocity for the flow:

$$u_{ave} = \frac{\int_0^h u_x dy}{h} = -\frac{1}{2\mu} \frac{dp}{dx} \frac{h^2}{6} \quad (4.7)$$

We can then calculate the ratio between the maximum and average velocity which is:

$$\frac{u_{max}}{u_{ave}} = \frac{3}{2}$$

This reasoning is important since the Reynolds number for the Poiseuille flow is based on the average speed, channel height and fluid viscosity, all expressed in terms of lattice units.

$$Re = \frac{h \cdot u_{ave}}{\nu} \quad (4.8)$$

The average speed for the simulation can be set to grant a very low Mach number below the compressibility limit. It is generally a good idea for it to be less than 10% of the speed of sound in the lattice to limit compressibility effects.

The pressure gradient was modeled as a nodal force dependent on the average speed, the viscosity and the channel's height:

$$\frac{dp}{dx} = g_x = \frac{3 \cdot u_{ave} \cdot \nu}{\left(\frac{h}{2}\right)^2} \quad (4.9)$$

The LBM simulation was performed with various mesh densities and at several Reynolds numbers. At first we will focus on the BGK-SRT implementation.

For the BGK-SRT model, the nodal force can be included in the calculation of the macroscopic velocity taking into account the relaxation time:

$$\mathbf{u} = \frac{1}{\rho} \sum_{i=1}^9 f_i \mathbf{e}_i + \frac{\mathbf{g}_x \cdot \boldsymbol{\tau}}{\rho} \quad (4.10)$$

The fluid was initialized with zero velocity in the whole domain. Full-way bounce-back boundary conditions were imposed on the upper and lower walls. Periodic boundary conditions were imposed in the streamwise direction.

A convergence criterion has to be established in order to decide when the LBM code has converged "enough", meaning that further iterations do not yield a significant precision increase. This operation is made fairly easy thanks to the existence of the analytical solution. In particular we can define the error between numerical and analytical velocities along the y direction using the L^2 -norm:

$$\text{error} = \sqrt{\frac{1}{N_y} \sum_i \left(\frac{u_{x,i} - u_{x,analytical,i}}{u_{max}} \right)^2} \quad (4.11)$$

The error is calculated every 2000 time steps of the simulation and when the difference between two consecutive errors becomes less than 10^{-6} the steady state solution is considered as reached. This convergence criterion yields very good results for various meshes and Reynolds numbers, providing a good trade-off between a reasonable precision and a good time management (the code isn't allowed to run for too much time with a negligible precision increase in the solution). Qualitative indications concerning the running time of the developed code (for this and all other test cases) are provided in Appendix D.

The general results in terms of streamwise velocity are shown in Figures 4.2 and 4.3 for $Re=200$ and a number of nodes along the vertical direction of $N_y = 101$. The analytical solution is expressed in the non-dimensional world and the streamwise speed is also rendered dimensionless with respect to the imposed average value of velocity ($u_{ave,lb} = 0.05$ lattice units/time step).

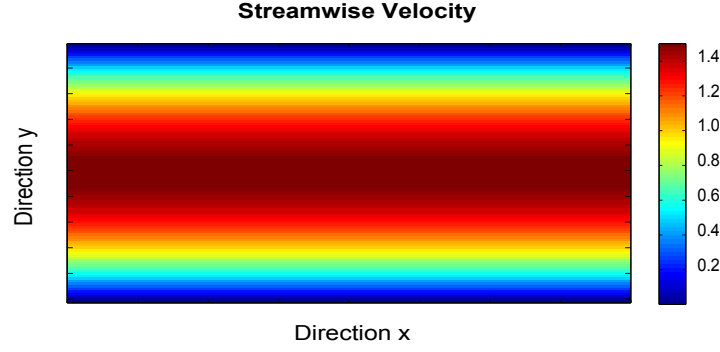


Figure 4.2 Streamwise velocity

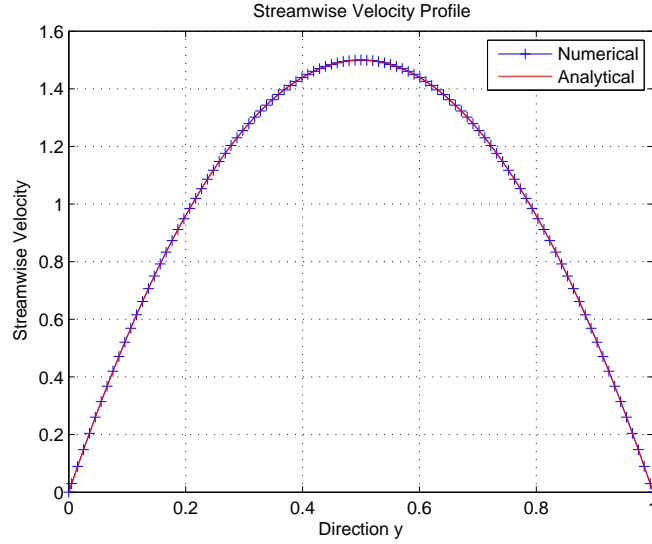


Figure 4.3 Streamwise velocity profile comparison between LBM simulation and analytical solution

Since the solution is steady-state, it is the same no matter the position along the channel at which the results are plotted. We observe excellent agreement between the LBM simulation and the analytical solution.

We can determine the order of convergence for this particular configuration using the procedure described in Section 4.1. In this case, the only significant direction is the vertical one. In fact an increased discretization in the horizontal direction doesn't improve the results because of the periodic boundary conditions imposed in that direction. The node number in

the x direction is therefore irrelevant to the results. The three increasingly finer meshes have the characteristics presented in Table 4.1.

Table 4.1 Poiseuille flow: grid characteristics

Mesh Type	Number of Nodes	Parameter r
3. Coarse	31	
2. Medium	61	$r_{21}=2.00$
1. Fine	121	$r_{21}=2.00$

For the following analysis, the parameter ϕ which is being analyzed is obviously the stream-wise velocity along the vertical direction at any point in the channel's length. The Reynolds number is fixed at $Re=400$ and the average speed of the profile is set as equal to 0.05 lu/ts, constant for the three mesh densities. The viscosity in lattice units is therefore the parameter which changes according to the mesh density in order to keep the Reynolds number constant. We will test four different configurations: the MRT and SRT collisions with bounce-back and Zou-He boundary conditions. In particular the Zou-He conditions will be used with zero velocity imposed on the boundary, in order to compare their performance to the bounce-back. The pressure driven flow is not particularly challenging in terms of stability and so the MRT method doesn't have significant advantages under this point of view. It is however interesting to check how the SRT and MRT collisions compare in terms of accuracy.

The convergence order for a generic flow is determined by the lowest convergence order of the different components employed: as stated previously the LBM for internal nodes is of second order, while the bounce-back boundary conditions are of first order. We should then expect a first order convergence for the bounce-back no matter the collision implemented. The Zou-He with null velocity imposed condition appears from literature to be of second order.

Results in terms of convergence order (p), extrapolated error (e_{ext}^{21}), fine grid error (e_a^{21}) and Fine Grid Convergence Index (GCI) are shown in Table 4.2.

Table 4.2 Poiseuille flow: convergence parameters

Collision	Boundary Condition	p	e_a^{21}	e_{ext}^{21}	GCI
SRT	Bounce-back	1.17	2.00%	1.77%	2.38%
SRT	Zou-He	1.89	0.085%	0.030%	0.038%
MRT	Bounce-back	1.17	2.00%	1.77%	2.38%
MRT	Zou-He	2.30	4.24e-05%	2.22e-05%	2.77e-05%

We can see that the LBM with bounce-back boundary condition is first order accurate, whilst the LBM with Zou-He boundary is second order accurate. With bounce-back boundary conditions the difference between the two finer meshes is in the order of 2 %. This difference decreases when we consider the difference between the extrapolated and finest mesh, meaning that, for an increasingly finer grid, the difference between successive refinings is diminishing. Accuracy comparisons between SRT and MRT for different boundary conditions are given in Figures 4.4 and 4.5.

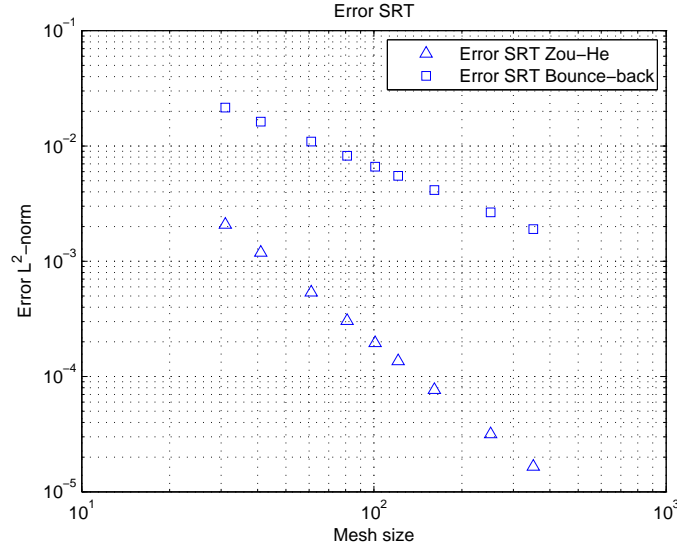


Figure 4.4 Logarithmic error plot for SRT collision

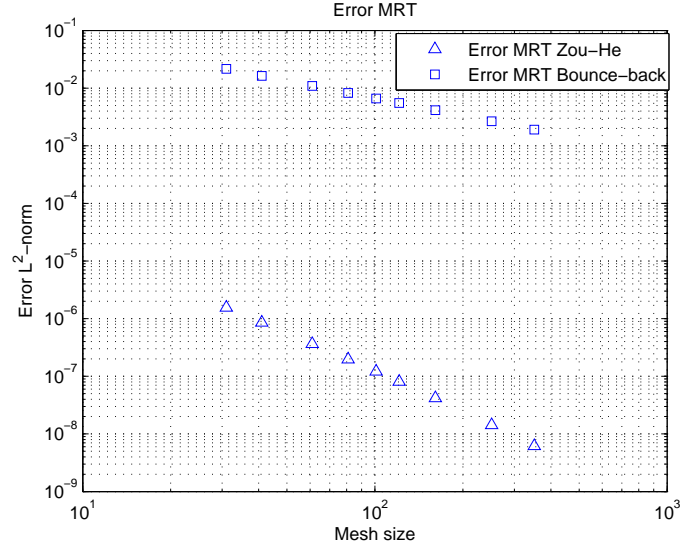


Figure 4.5 Logarithmic error plot for MRT collision

We notice that, for equal convergence criteria, the MRT method reaches a higher precision with respect to SRT, since the error with respect to the analytical solution is lower. As expected the Zou-He boundary conditions enable a decrease of the error at a greater pace with respect to the full-way bounce-back.

4.3 Couette-Poiseuille Channel Flow

Fluid motion between two infinite flat parallel plates, one of which is moving with a certain speed in the streamwise direction, is called Couette flow, from the name of the French physicist Maurice Couette. If we combine such flow with the pressure driven Poiseuille flow, we have the "Couette-Poiseuille" stream in which the fluid motion is influenced by both the moving lid and the pressure gradient. We can obtain the analytical solution to this type of flow by combining the two solutions for the two separate flows, or by solving the N-S equations for boundary conditions directly accounting for both the moving lid and the pressure gradient.

Figure 4.6 shows the geometry for a laminar Couette-Poiseuille flow of a Newtonian incompressible fluid between two flat plates separated by the distance h .

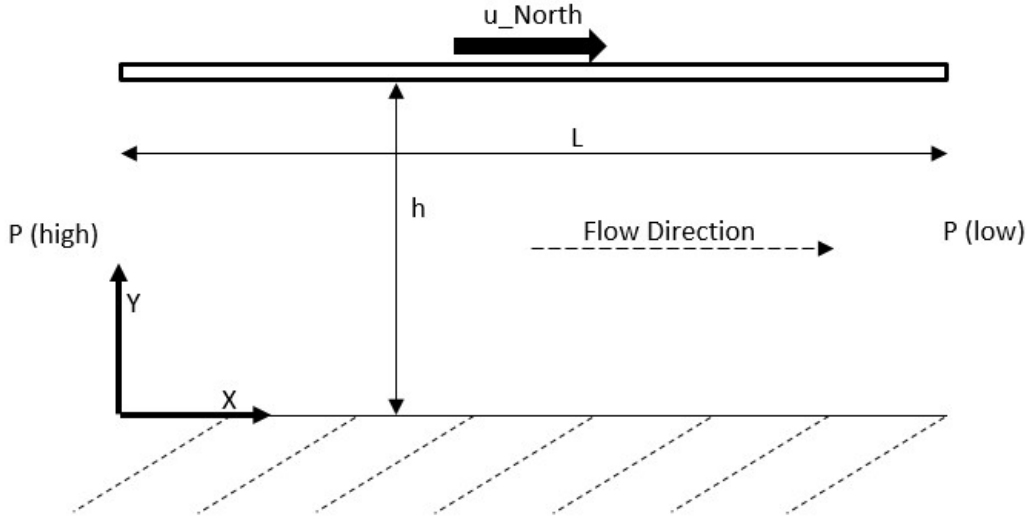


Figure 4.6 Geometry for the Couette-Poiseuille Flow

The boundary conditions for this test case will be:

$$\begin{cases} \text{at } y = 0 & \rightarrow & u_x = u_y = 0 \\ \text{at } y = h & \rightarrow & u_x = u_{North} \quad u_y = 0 \end{cases} \quad (4.12)$$

The well-known analytical solution for the velocity profile will be:

$$u_x(y) = \frac{u_{North}y}{h} - \frac{1}{2\mu} \frac{dp}{dx} (hy - y^2) \quad (4.13)$$

The convergence criterion is defined in the same way as the previous Poiseuille case, with the same imposed convergence. Considering the Reynolds number described in Equation 4.8, the LBM simulation for this case were run for $Re=300$ and a mesh size in the vertical direction of 61 nodes. The average speed of the Poiseuille profile was imposed as 0.05 lu/ts and the speed of the moving lid as $u_{North} = 0.01$ lu/ts. Periodic conditions were imposed in the streamwise direction. On the lower boundary a full-way bounce-back boundary condition was implemented, while on the upper wall the missing distributions were calculated with the Zou-He velocity boundary condition.

Figure 4.7 shows the results comparison between simulations run with the BGK-SRT and MRT collisions and the analytical velocity profile. Lengths are rendered nondimensional to obtain a unitary channel height. The velocity is nondimensionalized with respect to the average velocity of the Poiseuille profile.

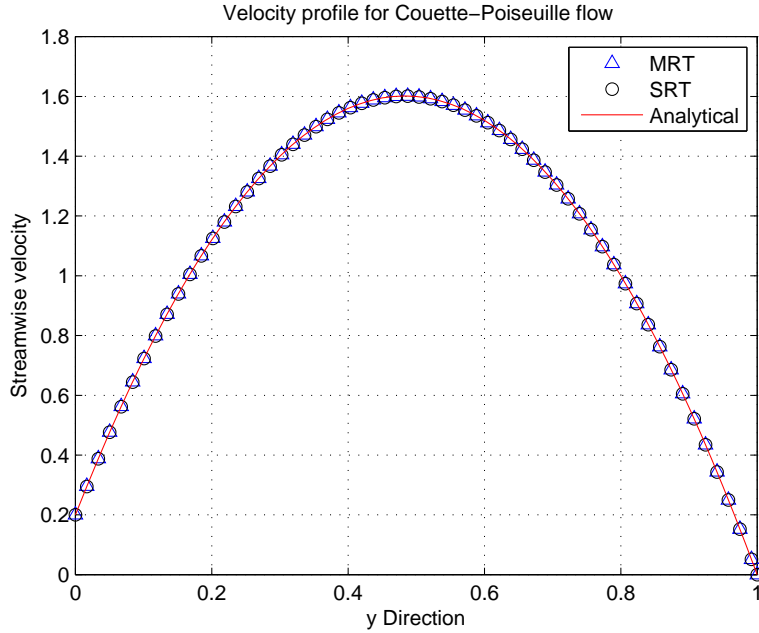


Figure 4.7 Velocity for the Couette-Poiseuille flow along vertical direction

Excellent agreement is observed between the analytical speed profile and the profiles computed with the developed LBM code, both with BGK-SRT and MRT collisions.

4.4 Lid-Driven Cavity Flow

One of the most interesting fluid-dynamics problems is for sure the lid-driven cavity flow. In this section we will describe and discuss this particular case checking whether there is a good agreement between the developed LBM solver and results available in literature.

The simulation of the incompressible flow in a closed enclosure driven by the top lid's translation is for sure one of most tested and studied cases in fluid dynamics. In fact, often, even Navier-Stokes based solvers are tested for this specific application since the flow is very complex and includes vortex formations. The first comprehensive study ever performed was the work of [15] which has been the benchmark for many decades. For what concerns the LBM, a comprehensive analysis of the lid driven cavity flow was performed by [19].

The considered cavity is square and the flow is driven by the top lid which moves eastwards with uniform speed. The discretisation on each of the sides was varied up to 500 nodes per side, resulting in a total of 250000 nodes. Following [15] work, the Reynolds number was varied up until 10000. For this type of geometry, the Reynolds number is based on the length of the moving lid, its speed, the fluid's viscosity and density following the expression:

$$Re = \frac{\rho u_{max} L}{\mu} = \frac{u_{max} L}{\nu} \quad (4.14)$$

In the LBM simulation all the parameters are in lattice units and the uniform top lid velocity is set to be $u_{max} = 0.05$ lu/ts to limit compressibility effects. A sketch of the geometry is shown in Figure 4.8

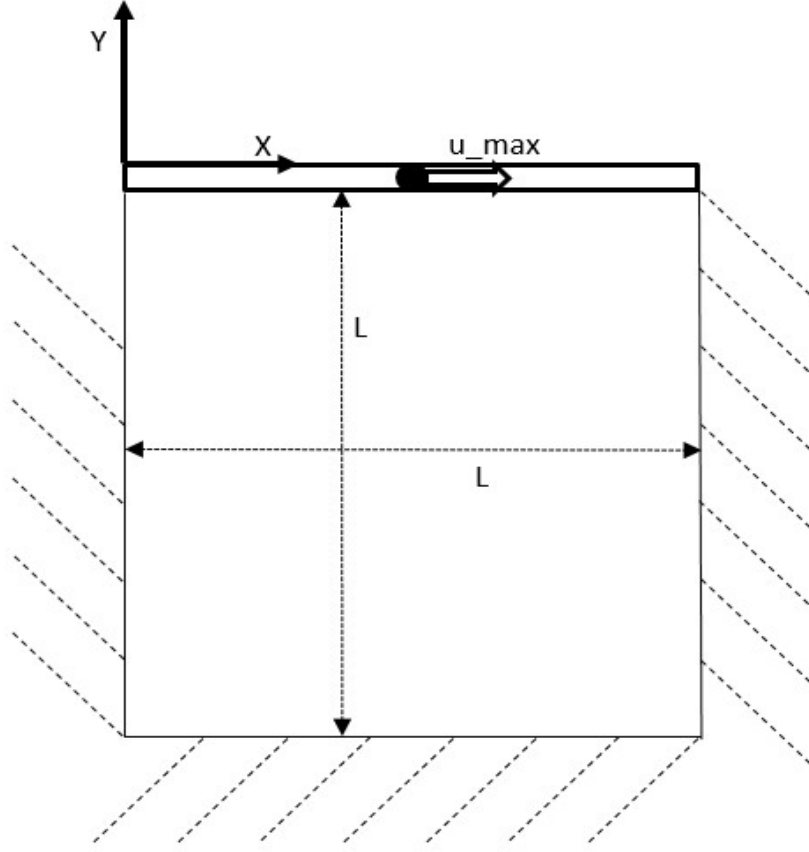


Figure 4.8 Geometry of the lid driven cavity

Initially the fluid, with unitary density, is at rest, with null speed on every node except the ones belonging to the top lid. Distribution functions are initialized by setting them as equal to the equilibrium distributions calculated for the imposed velocity and density distributions. We should then allow many iterations for the solution to evolve towards the steady state. Here an issue arises: how do we define the steady state numerically given that an analytical solution for this flow is not available? A common approach is to set a convergence criterion relying on either an imposed maximum iterations number or the comparison between successive time steps. The first approach is not ideal since the number of iterations required for the evolution towards the steady state vary according to the Reynolds number and mesh. The second approach was therefore chosen: the mean total speed (defined in Equation 4.15) along the vertical centerline was computed every 2000 iterations. The reaching of the steady state was then imposed as the time when the difference between two successive mean values was lower than 10^{-8} . This convergence criterion shows good results in terms of precision and reasonable computation time for all of the configurations tested.

The bounce-back boundary condition is imposed on the three fixed walls, while at the top Zou-He conditions calculate missing distributions for the moving boundary, taking the lid's speed into account. The upper corners are singularities, since we should decide whether they belong to the eastern and western wall, or are part of the moving lid. After a few tests in which both configurations were tested, the second solution was chosen, although no significant difference was actually observed between the two results.

The results are expressed in terms of total speed:

$$u_{total} = \sqrt{u_x^2 + u_y^2} \quad (4.15)$$

Lengths are normalized with respect to the side's length and speeds are nondimensionalized with u_{North} . This procedure is performed to allow comparison between the present work and that of [15], which is expressed in terms of a unitary length square cavity with a unit velocity of the top boundary.

Figure 4.9 gives a general idea of the total speed in a cavity at $Re=1000$. We clearly notice the presence of a central vortex. Even if not visible in this figure, recirculation areas at the bottom corners are also present. Their size and location varies a lot according to the Reynolds number of the flow. If we look at the streamlines of the flow at various Reynolds numbers, the vortices are more clearly visible, as shown by Figures 4.10, 4.11, 4.12 and 4.13. It should be noted that a SRT collision was chosen for the lower Reynolds of 100 and 400, while a MRT collision was preferred for higher Re . This is because the SRT method becomes unstable for a Re of around 2000. The higher stability of the MRT collision is consistent with the observations by [24].

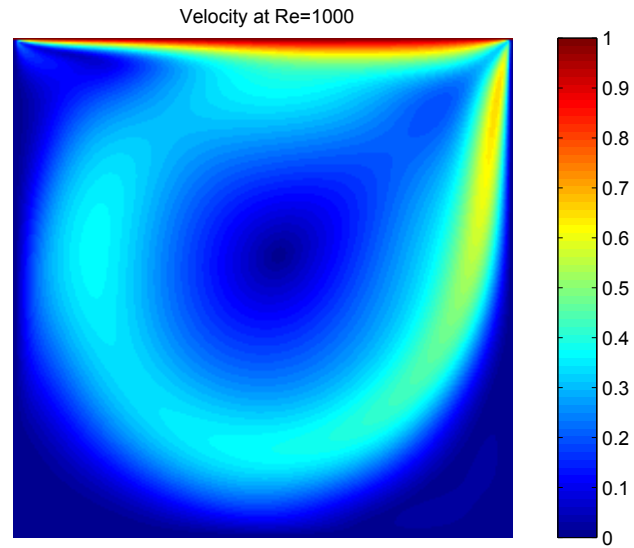


Figure 4.9 Total Velocity profile at $Re=1000$

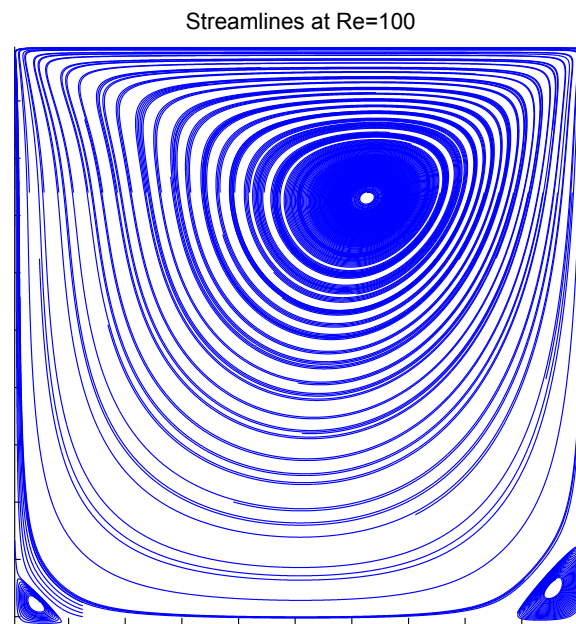


Figure 4.10 Streamlines at $Re=100$

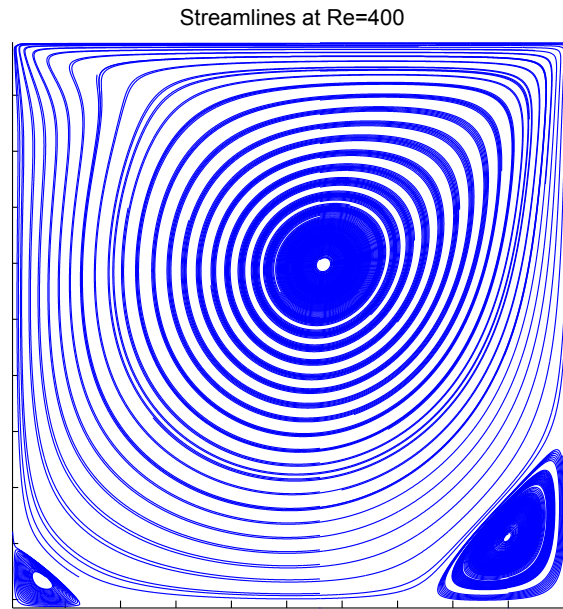


Figure 4.11 Streamlines at $Re=400$

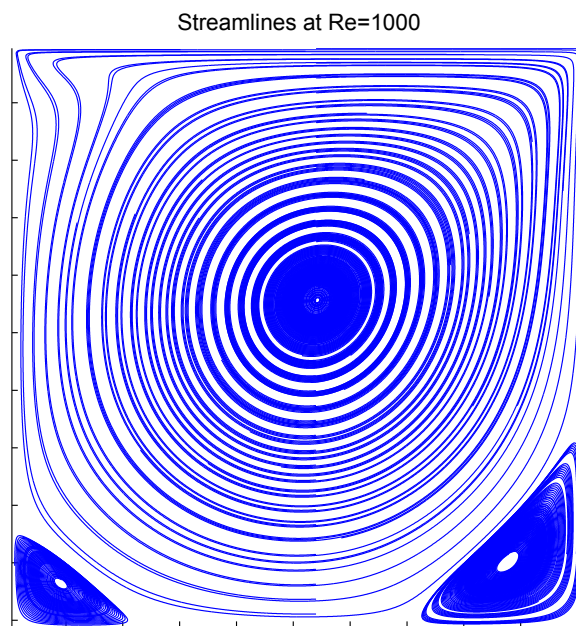


Figure 4.12 Streamlines at $Re=1000$

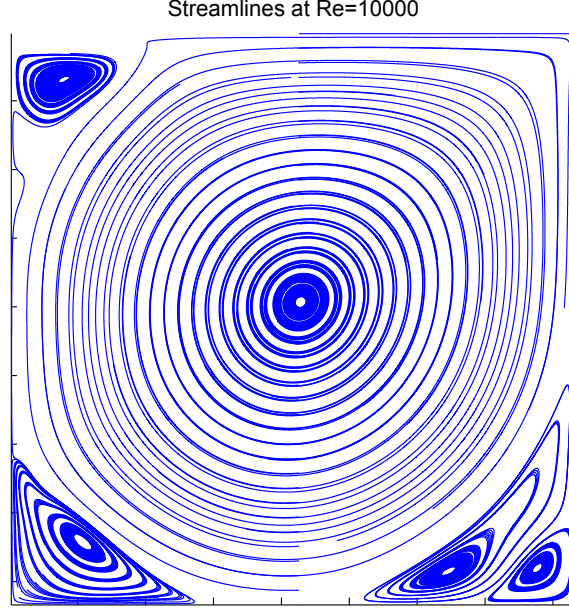


Figure 4.13 Streamlines at $Re=10000$

We notice that, as the Reynolds increases, the bottom vortices gain more and more importance. Also, at $Re=10000$, secondary vortices appear in the bottom corners and a top left corner vortex develops. In order to rigorously validate the developed LBM code simulating cavity flow, we can compare the vortices' locations with the work of [15]. Results are given in Table 4.3, for Reynolds of up to $Re=1000$.

Table 4.3 Primary and bottom corners' vortex positions at varying Reynolds compared with results from [15]

Vortex Positions	Method	Mesh	Primary Vortex		Lower Left		Lower Right	
	x	y	x	y	x	y	x	y
<i>Re</i> 100	Ghia et al.	129x129	0.6172	0.7344	0.0313	0.0391	0.9453	0.0625
<i>Re</i> 100	LBM SRT	203x203	0.6109	0.7358	0.0341	0.0394	0.9461	0.0591
<i>Re</i> 400	Ghia et al.	257x257	0.5547	0.6055	0.0508	0.0469	0.8906	0.125
<i>Re</i> 400	LBM SRT	203x203	0.5559	0.6059	0.0591	0.0443	0.8892	0.1280
<i>Re</i> 1000	Ghia et al.	129x129	0.5313	0.5625	0.0859	0.0781	0.8594	0.1094
<i>Re</i> 1000	LBM MRT	203x203	0.5320	0.5616	0.0837	0.0788	0.8572	0.1083

At $Re=10000$ [15] observe the presence of secondary and tertiary vortex at the bottom corners. In the corresponding Figure 4.13 however the tertiary ones are not visible, because they are very small and very close to the walls. Table 4.4 shows that even for this case the LBM is able to locate the primary and secondary vortices' positions in the cavity very close to the values given in the reference.

Table 4.4 Primary and bottom corners' vortex positions at $Re=10000$ compared with results from [15]

Vortex Positions	Method	Mesh	Primary Vortex		Lower Left		Lower Right	
			x	y	x	y	x	y
Re 10000	Ghia et al.	257x257	0.5117	0.5333	0.0586	0.1641	0.7656	0.0586
Re 10000	LBM MRT	417x417	0.5156	0.5262	0.0795	0.1103	0.7681	0.0598
	Method	Mesh	Top Left Vortex		2^{nd} Lower Right			
			x	y	x	y		
Re 10000	Ghia et al.	257x257	0.0703	0.9141	0.9336	0.0625		
Re 10000	LBM MRT	417x417	0.0887	0.9171	0.9328	0.0623		

We can also compare the velocity profiles obtained via the LBM simulation to the results of [15] for Re numbers of 100, 400, 1000 and 10000. In particular we will focus on the velocity profiles obtained on the vertical and horizontal centerlines. On the former, the horizontal velocity profile, $u_x(y)$, is provided, while on the latter the vertical velocity profile $u_y(x)$ is reported. The LBM simulations are performed with the same Re number and collision model indicated in Tables 4.3 and 4.4. Figures 4.14, 4.15, 4.16, 4.17 show the velocity profiles on the vertical and horizontal centerlines for Re numbers of 100, 400, 1000 and 10000.

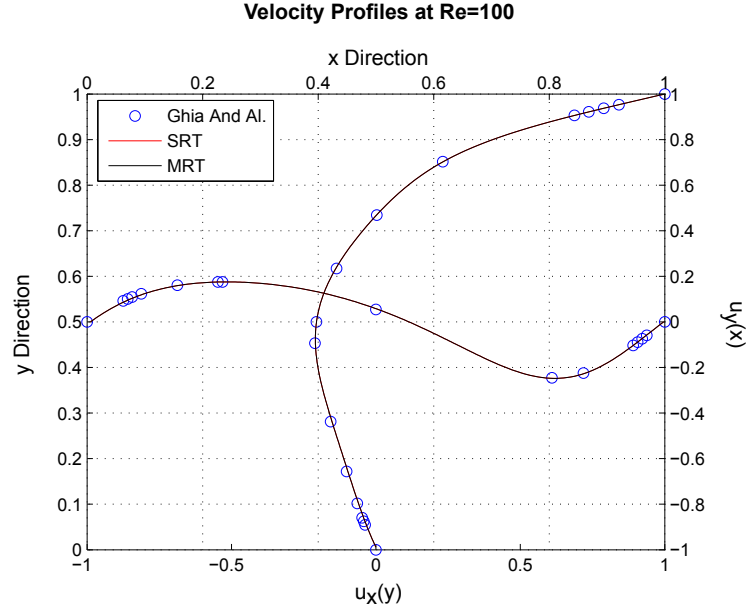


Figure 4.14 Comparison between BGK-SRT collision, MRT collision and [15] results for $Re=100$

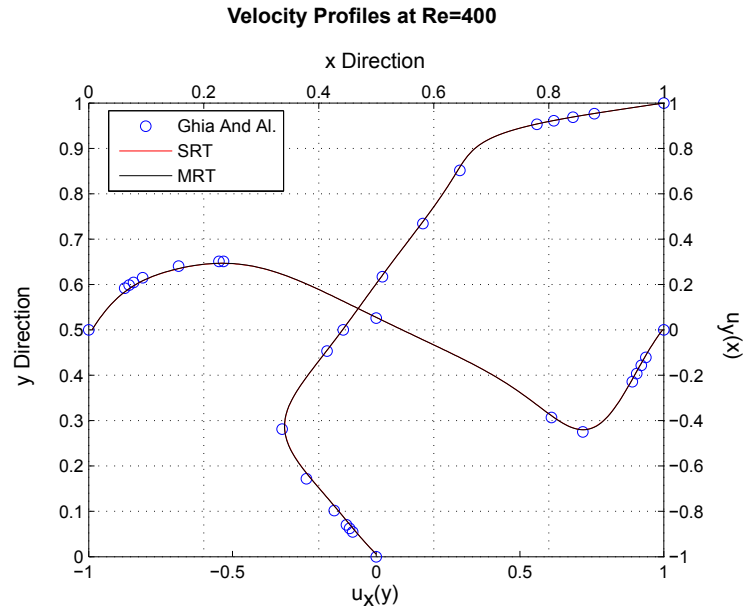


Figure 4.15 Comparison between BGK-SRT collision, MRT collision and [15] results for $Re=400$

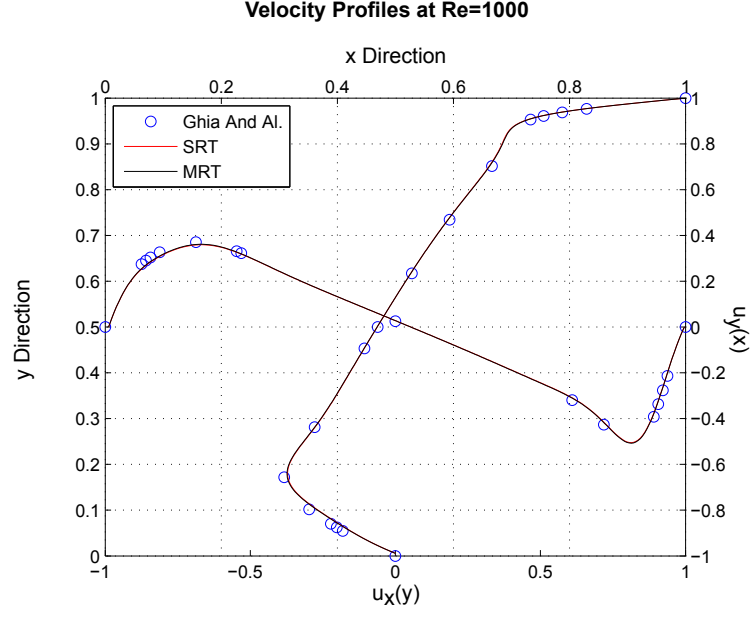


Figure 4.16 Comparison between BGK-SRT collision, MRT collision and [15] results for $Re=1000$

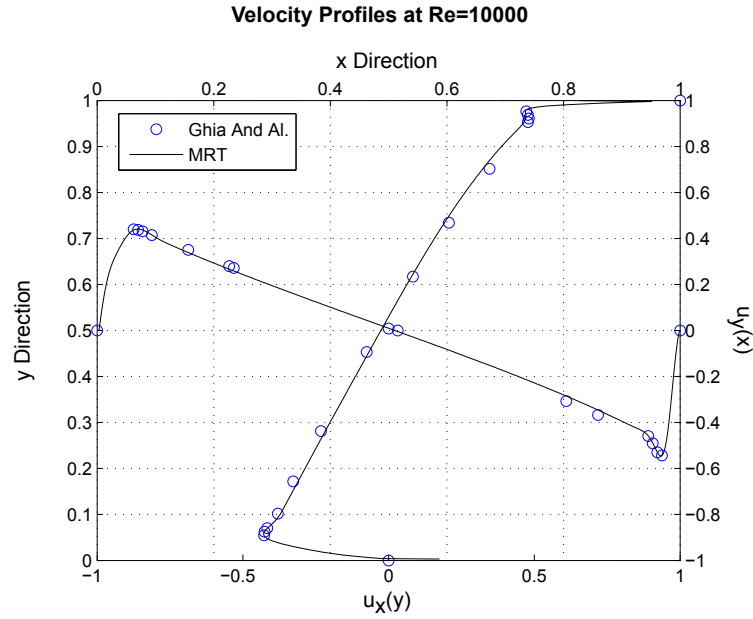


Figure 4.17 Comparison between BGK-SRT collision, MRT collision and [15] results for $Re=10000$

This particular test case combines both the Zou-He velocity boundary condition and the bounce-back condition and so it is interesting to check the order of accuracy and the extrapolated profile using the procedure presented in Section 4.1. Simulations are performed with a Re number of 400 and a top lid velocity of 0.05 lu/ts. The data of the three increasingly fine grids are reported in Table 4.5.

Table 4.5 Lid-driven cavity: grid characteristics

Mesh Type	Node Number	Parameter r
3. Coarse	129x129	
2. Medium	257x257	$r_{32} = 2$
1. Fine	513x513	$r_{21} = 2$

The two reference parameters which can be used to measure the convergence and analyze the mesh error are the speed profiles on the horizontal and vertical centerlines.

Table 4.6 Lid-driven cavity: convergence parameters

Collision	Location	ϕ Parameter	p	e_a^{21}	e_{ext}^{21}	GCI
SRT	Horizontal centerline	$u_y(x)$	0.9721	2.20%	2.24%	2.89%
MRT	Horizontal centerline	$u_y(x)$	0.9124	2.02%	2.20%	2.84%
SRT	Vertical centerline	$u_x(y)$	0.9920	1.83%	1.88%	2.45%
MRT	Vertical centerline	$u_x(y)$	0.9901	1.74%	1.85%	2.41%

As expected we have a first order accuracy for increasingly fine meshes, because of the presence of the first order bounce-back boundary condition. The differences between the intermediate and refined meshes' results and between the fine and extrapolated results are very small. This indicates that the chosen meshes are sufficiently fine to correctly capture the flow characteristics. The low GCI also supports this conclusion.

We can also calculate the extrapolated values for the velocity profiles and compare them to

the finest mesh's results. Figures 4.18 and 4.19 show that the extrapolated results are even closer to the results of [15] with respect to the results obtained with the finest mesh.

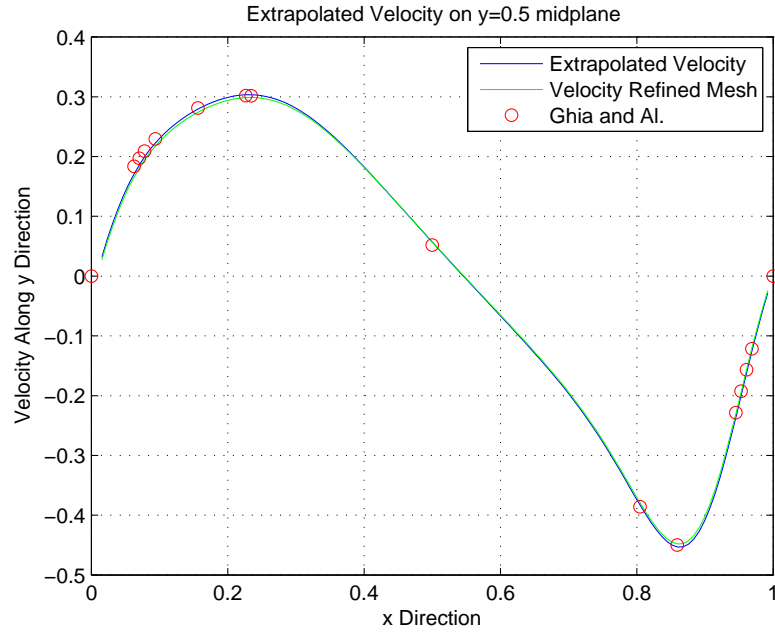


Figure 4.18 Comparison between extrapolated velocity profile, velocity on most refined mesh and experimental solution by [15] for horizontal centerline and $u_y(x)$

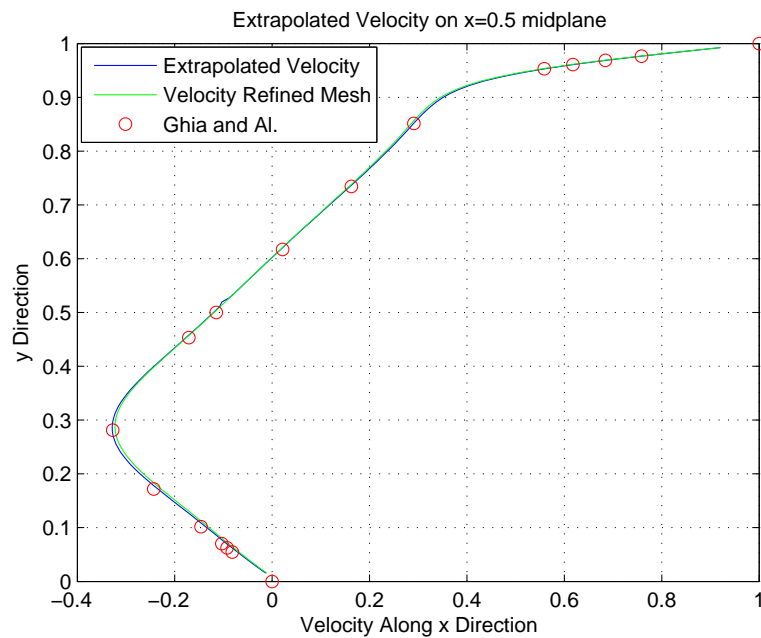


Figure 4.19 Comparison between extrapolated velocity profile, velocity on most refined mesh and experimental solution by [15] for horizontal centerline and $u_x(y)$

4.5 Unsteady Flow Past Square Obstacle

It is interesting to verify if the LBM is able to correctly simulate unsteady flows. In particular, since not many simulations can be found in literature concerning the unsteady flow past a square obstacle in a duct, it was decided to implement this test case.

Depending on the flow's Reynolds number, several flow regimes can occur for this test case. The Reynolds number is based on the obstacle's diameter, a reference velocity and the fluid's viscosity. For very small Reynolds numbers, no flow separation occurs since the viscous forces keep the flow attached to the obstacle. As the Reynolds number increases to the order of 5-10, detachment occurs behind the obstacle giving origin to two symmetrical vortices which form a closed recirculation area whose size increases with the Reynolds number. Then, at a Reynolds number of around 60 (some controversy does exist in literature on the exact value [5]), Von Kármán periodic vortices are formed. In fluid dynamics, a von Kármán vortex street is a repeated pattern of vortices caused by the flow's detachment around blunt bodies. In nature, it was first observed and investigated because of the so called "singing" of suspended telephone or power lines.

It is generally accepted [13] that, for $Re < 300$, the vortex shedding is mainly a 2D problem: after this value the vortices acquire clear 3D characteristics. Since the aim of this validation test case is limited to 2D, only Reynolds numbers well below such limit will be tested.

The vortex shedding case will be analyzed by comparing our LBM implementation to the data obtained via a 3D Finite-Volume Method by [5]. In this case, a confined flow past a square cylinder is simulated given a blockage ratio, defined as the cylinder's diameter over the channel's height, of $1/8$. The definition of the geometry is given in Figure 4.20 and Equation 4.16. We will compare our LBM results to the FVM results obtained by [5] for several parameters such as drag coefficient, recirculation length and velocity profiles for both the steady and unsteady cases. The FVM values presented in the following figures were obtained from [5]'s work using Plot Digitizer, a tool available online at [38].

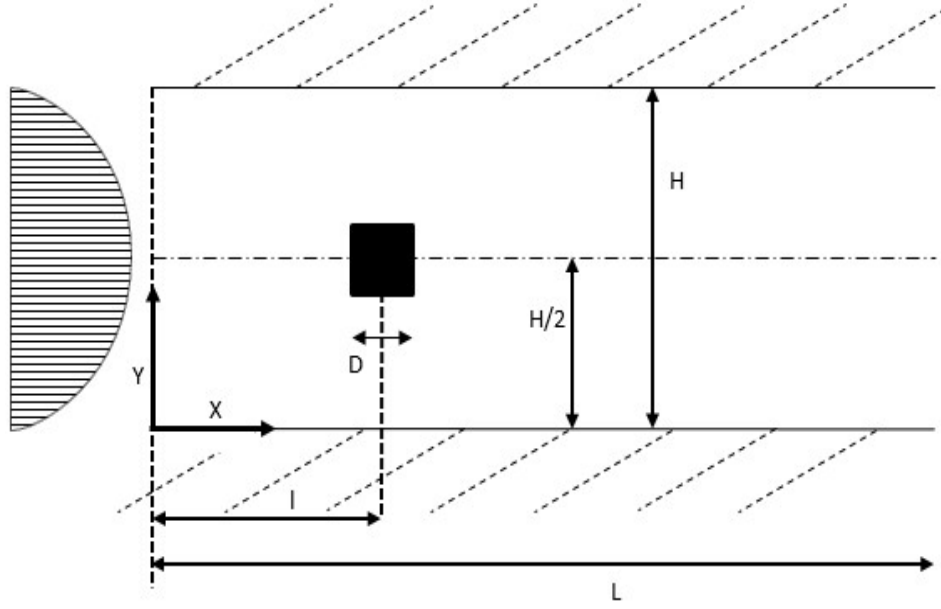


Figure 4.20 Geometry (not in scale) of flow past a square obstacle

$$\left\{ \begin{array}{l} \text{Inlet: Parabolic Profile} \quad u(y) = 4 \frac{u_{max}}{H^2} (y \cdot H - y \cdot y) \\ \frac{D}{H} = \frac{1}{8} \\ \frac{L}{D} = 50 \\ l = \frac{L}{5} \end{array} \right. \quad (4.16)$$

The LBM simulation was performed on a regular orthogonal lattice with the BGK-SRT collision operator. The collision in moment space (MRT) was also tested. It yields very similar results to the SRT and, since in this case stability is not really an issue, as simulations are run at low Reynolds numbers with a very fine grid, it was judged sufficient to provide the BGK-SRT based results. The maximum velocity of the inlet profile, u_{max} , was set to be 0.05 lu/time step, well below the point at which compressibility effects become important. Several different meshes were tested, always maintaining the ratios indicated in Equation 4.16. Then, because of the good trade-off between a reasonable calculation time (a few hours) and a good accuracy, a mesh with 2000 nodes in the x direction and 320 nodes on the y direction was

chosen. The fluid viscosity varies according to the Reynolds number, the maximum velocity and the diameter of the obstacle in lattice units, which is dependent on the number of nodes used to discretize the channel's length. The density of the fluid was, as always, assumed unitary.

$$Re = \frac{u_{max} \cdot D}{\nu} \quad (4.17)$$

The boundary conditions of the LBM code, a bounce-back boundary was imposed on the upper and lower walls and on the square cylinder perimeter. At the inlet a parabolic velocity profile was prescribed in order to simulate a fully developed laminar velocity profile. The corresponding distribution functions were calculated with the Zou-He procedure. At the exit, a constant pressure profile is imposed but, given the existing relation for LBM simulations between pressure and density given in Equation 3.20, this is equivalent to imposing a density. Once the density is imposed the correspondent velocity profile can be calculated with the procedure summarized in Section 3.2.6 and then the Zou-He conditions can be applied.

The initial conditions in terms of velocity were set using the same parabolic velocity profile as the inlet on the whole domain (except for the obstacle and walls where speed is of course null). This is done to enhance the convergence and to avoid having a long transitory time needed for the profile's development. Given the assigned speed, the initial distribution functions were calculated by imposing them as equal to the equilibrium distributions related to the velocity initialization.

In Figures 4.21 and 4.22, we can observe the velocity distribution for $Re=10$ and $Re=100$. The first case is clearly steady with the closed recirculation zone just behind the obstacle. At $Re=100$ we observe the formation of the periodic Von Kármán vortices. The results are shown for a time step sufficiently long after the simulation's start so that the initial transitory period is over. The aim of the figures is to provide a qualitative idea of the flow past an obstacle test case.

Velocity for x direction at $Re=10$

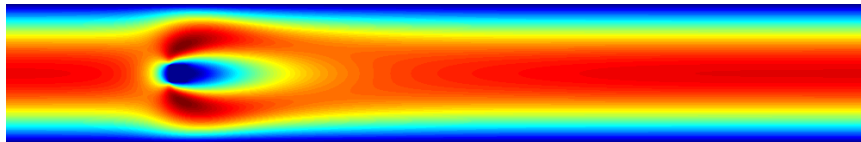
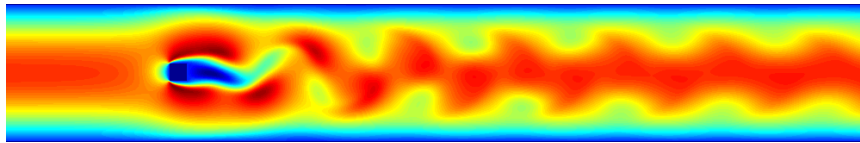
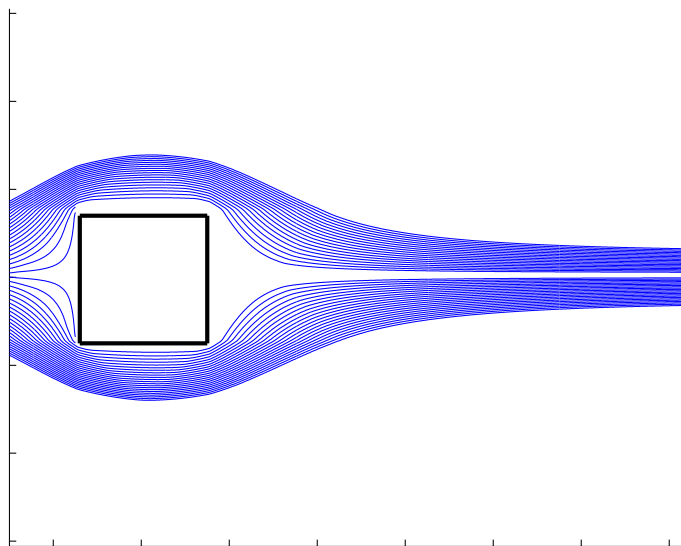


Figure 4.21 Streamwise velocity at $Re=10$

Velocity for x direction at $Re=100$ Figure 4.22 Streamwise velocity at $Re=100$

To have a clearer qualitative idea of the flow's characteristics for this test case, we can also observe the streamlines in the area close to the cylinder. Figures 4.23, 4.24, 4.25 show the streamlines at three different Reynolds numbers. In Figure 4.23, at $Re=1$, we notice that the flow remains adherent to the obstacle. In Figure 4.24, at $Re=10$, we notice the closed recirculation area and, in Figure 4.25, at $Re=100$, we observe the Von Kàrmàn vortices.

Streamlines at $Re=1$ Figure 4.23 Streamlines at $Re=1$

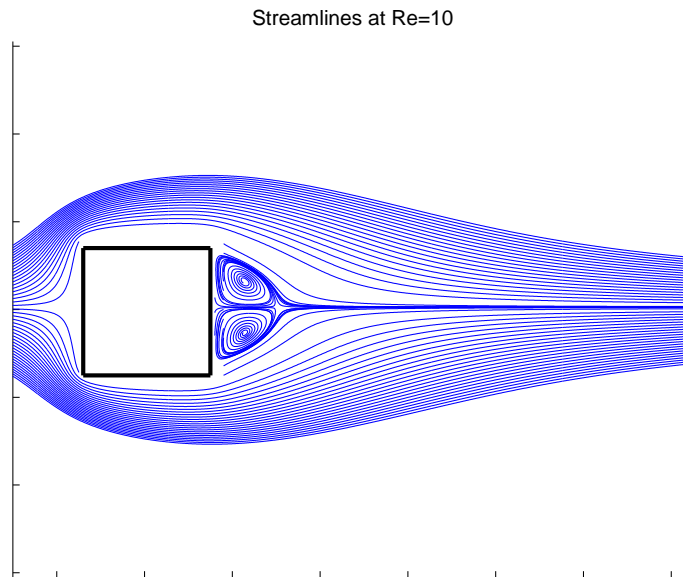


Figure 4.24 Streamlines at $Re=10$

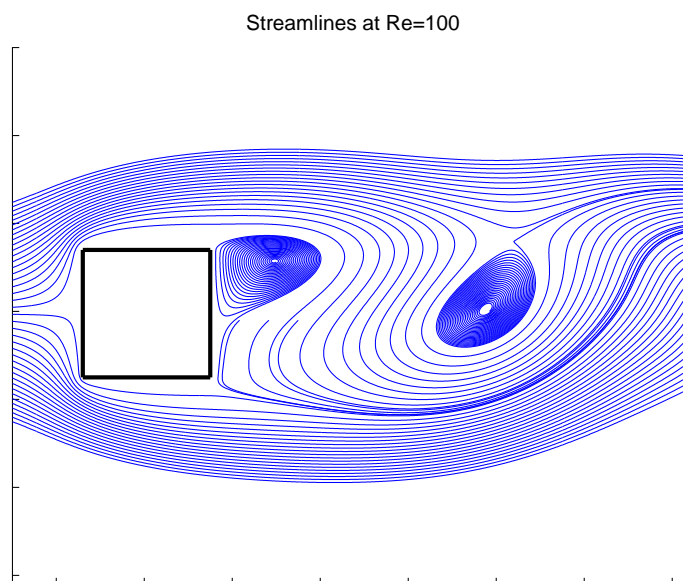


Figure 4.25 Streamlines at $Re=100$

For the steady case, we will investigate two important parameters: the recirculation length

(L_r) and the drag coefficient (C_d). For the former, which could also be defined as the flow's reattachment length, an empirical equation exists, obtained by linear interpolation of [5]'s results. It is valid for a blockage ratio of 1/8 and for Reynolds numbers between 5 and 60:

$$\frac{L_r}{D} = -0.065 + 0.0554Re \quad \text{Valid for } 5 < Re < 60 \quad (4.18)$$

The measuring of the recirculation length for the LBM is a little approximate, since it is hard to define an exact condition verified for the exact point where the flow is reattached. Results however are excellent and follow almost perfectly the linear trend dictated by the empirical relation.

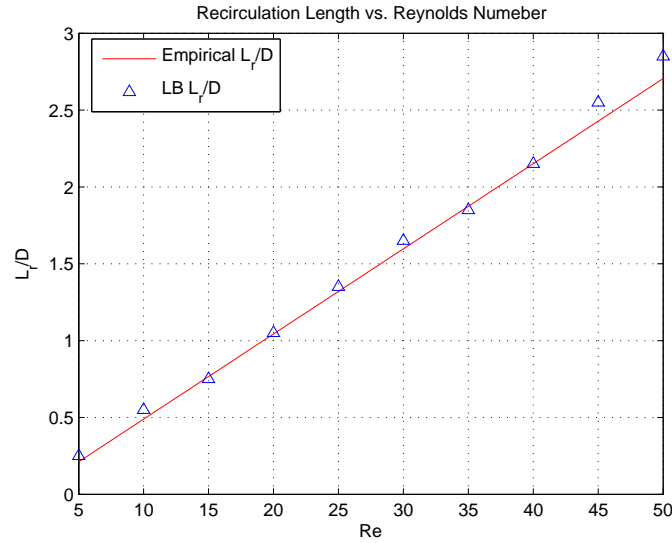


Figure 4.26 Normalized recirculation length vs. Reynolds Number

In fluid dynamics, the drag coefficient (C_d) is a dimensionless quantity used to quantify the drag or resistance of an object in a fluid environment in the direction of the flow. Together with the lift coefficient, which gives a non-dimensional idea of the forces in a direction normal to the flow's direction, it is often studied in fluid dynamics simulations. The following equations express the relationship between the total forces acting on a blunt body of characteristic size A and the lift and drag coefficients:

$$F_x = F_{\text{drag}} = \frac{1}{2}\rho u^2 C_d A \quad (4.19)$$

$$F_y = F_{\text{lift}} = \frac{1}{2}\rho u^2 C_l A \quad (4.20)$$

In the LBM, the coefficients can be calculated from the weighted summation of the distributions entering and exiting the nodes representing the square obstacle's boundary:

$$F_{x,i}(\text{boundary}) = e_{x,i} [f_i(\text{boundary}, t+1) + f_i(\text{boundary}, t)] \quad (4.21)$$

$$F_{y,i}(\text{boundary}) = e_{y,i} [f_i(\text{boundary}, t+1) + f_i(\text{boundary}, t)] \quad (4.22)$$

$$(4.23)$$

The total force acting on a single node will be equal to the sum of the directional forces:

$$F_{x,\text{total}}(\text{boundary}) = \sum_i F_{x,i}(\text{boundary}) \quad (4.24)$$

$$F_{y,\text{total}}(\text{boundary}) = \sum_i F_{y,i}(\text{boundary}) \quad (4.25)$$

The lift and drag total forces are related to the lift and drag coefficients by the equations presented in 4.20. For the 2D square obstacle flow, the characteristic length corresponds to the obstacle's diameter and so:

$$C_d = \frac{F_{x,\text{tot}}}{0.5\rho D u_{\text{max}}^2} \quad (4.26)$$

$$(4.27)$$

$$C_l = \frac{F_{y,\text{tot}}}{0.5\rho D u_{\text{max}}^2} \quad (4.28)$$

In the case of the steady flow past an obstacle, the lift coefficient turns out to be negligible since the flow case is symmetric and therefore no forces are generated in the vertical direction. The drag coefficient is, on the other hand, non null and strongly dependent on the Reynolds number. In Figure 4.27 we observe the comparison between the drag coefficient calculated with the LBM and the FVM results given in the reference, at different Reynolds. In this case

also, a very good agreement is observed. The small differences are mainly related to the fact that for the FVM the grid is more refined close to the obstacle, while the LBM simulation is performed on a uniform grid.

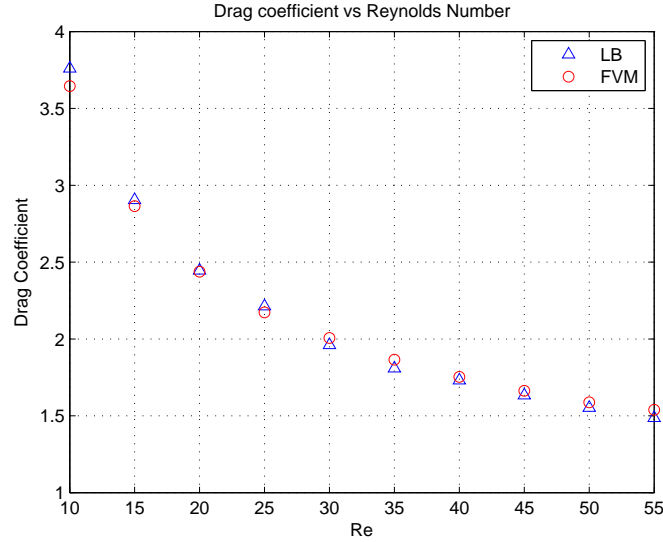


Figure 4.27 Comparison between LBM and FVM for Steady Drag coefficient

We will now shift our attention towards the unsteady flow simulation. In this case, because of the non stationary flow, comparison between FVM and LBM will be a little more complex. It is fundamental to find a time step at which to compare the two methods. This was done considering the centerline position at 41 % of the horizontal direction: the moment of comparison was chosen as the instant at which the vertical velocity in this point switches from negative to positive. This should allow a correct comparison between the two methods, since the definition of the position in a non-dimensional way allows for quick and precise identification of the location no matter the domain's size. Since the $Re=100$ flow has been shown to generate the Von Kármán vortices together with being low enough for the phenomenon to be 2D, it was chosen for the following analysis.

In Figures 4.28 and 4.29 we observe the comparison for the streamwise and cross-stream velocity along the x direction. In Figures 4.30 and 4.31 the streamwise and cross-stream velocity components are shown along the vertical direction at three different locations: 20%, 28% and 36% of the channels length L . It should be noted that the comparison is performed with non dimensional data. The FVM data were normalized according to the geometrical information only, since the reference speed used by [5] was already unitary. In the LBM the vertical and horizontal directions were normalized according to the number of nodes used in

the respective directions and the speed was normalized with respect to the maximum speed of the parabolic profile in lattice units. It should be noted that the FVM results were available for only a portion of the domain and were not given for the area closer to the outflow.

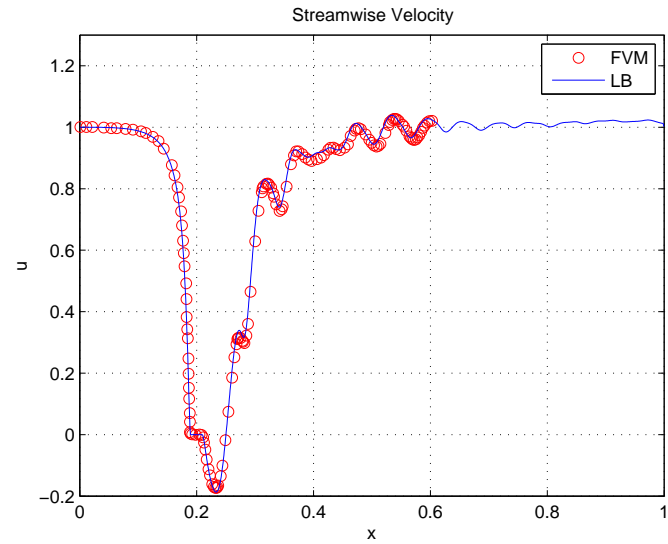


Figure 4.28 Streamwise velocity along the channel’s length

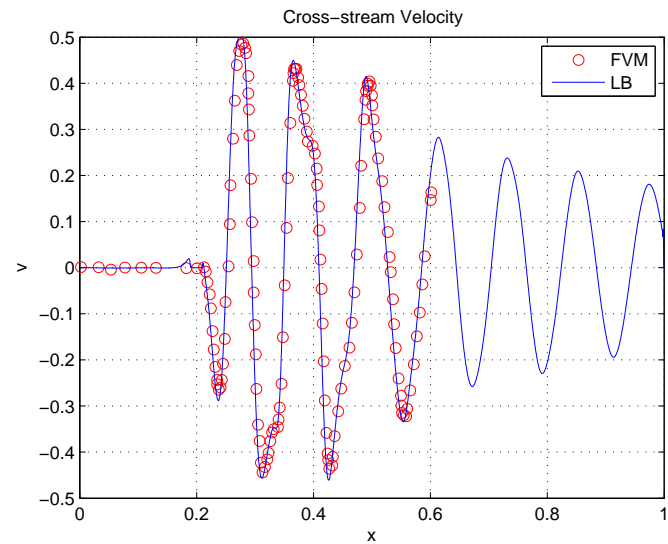


Figure 4.29 Cross-stream velocity along the channel’s length

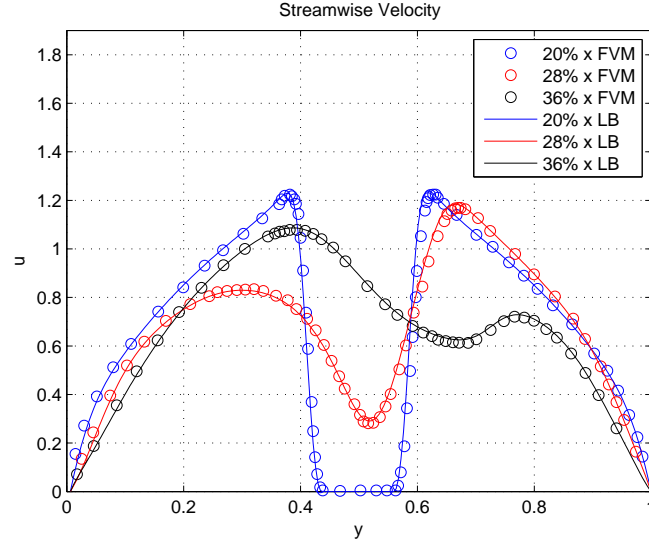


Figure 4.30 Streamwise velocity at three different locations in the channel's length

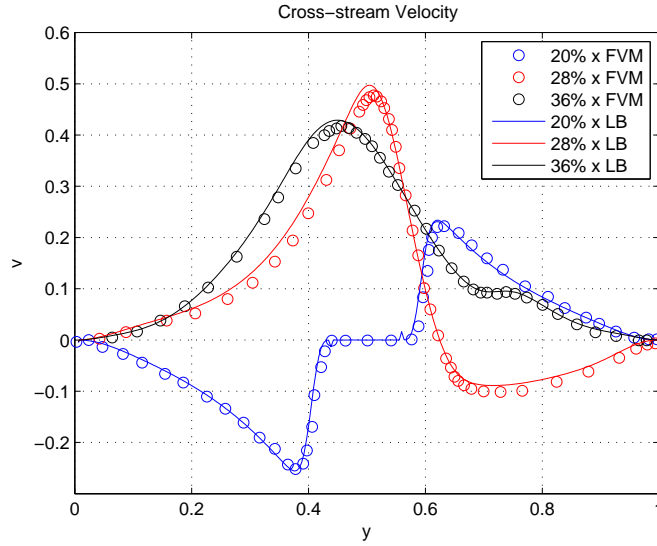


Figure 4.31 Cross-stream velocity at three different locations in the channel's length

In general, there is an excellent agreement between the FVM results and the LBM code. Due to the high unsteadiness of the solution, however, a very slight difference is observed in some of the cases, even if the general trend of the curves is maintained. Another possible source of difference is the grid refinement which was used in the FVM, but not in the LBM. This

causes more accurate FVM results in the area closest to the obstacle, where the grid is finer than the LBM one. On the other hand results of FVM are less accurate further away from the square cylinder since there the FVM grid is coarser than the LBM grid.

The lift and drag coefficients can be computed for the unsteady flow case also. Figures 4.32 and 4.33 show the qualitative behavior of the lift and drag coefficients over several iterations. We notice the periodic nature of the lift coefficient, which oscillates around a null value. The drag coefficient, on the other hand, presents some slight variations around a mean value, due to unsteadiness of the flow. The mean of the drag coefficient is shown in Figure 4.34 for various Reynolds numbers. The drag coefficient at steady Reynolds numbers is also plotted again, in order to appreciate its generic trend for increasing Re .

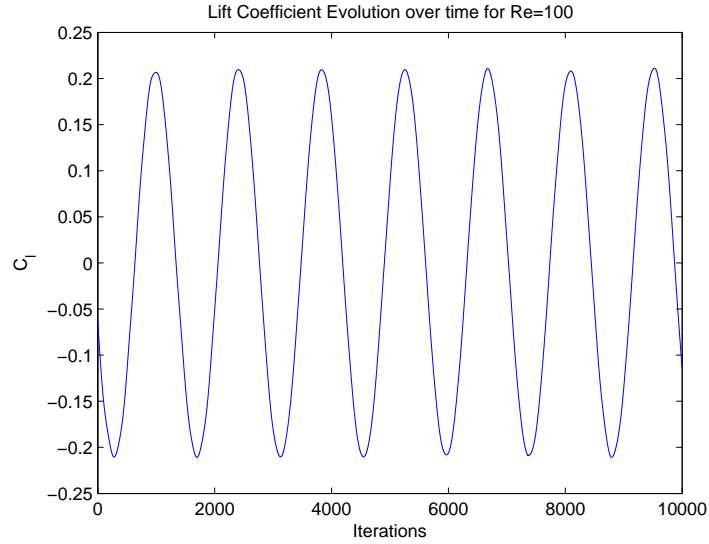


Figure 4.32 Lift coefficient evolution over time for $Re=100$

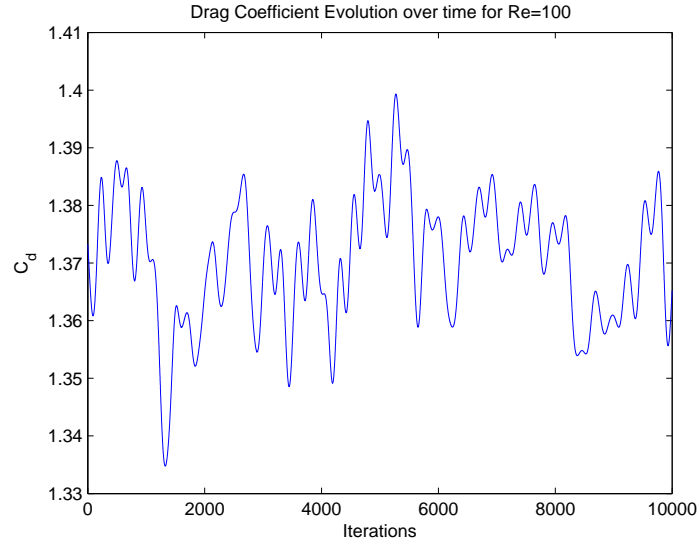


Figure 4.33 Drag coefficient evolution over time for $Re=100$

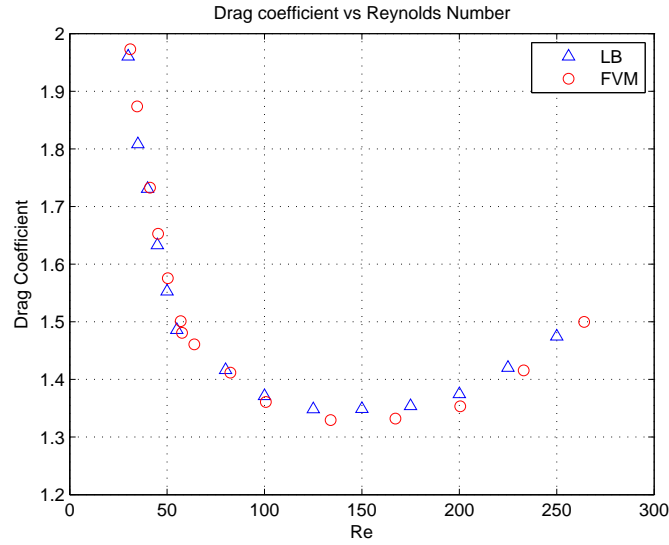


Figure 4.34 Drag Coefficient for steady ($Re < 60$) and unsteady ($Re > 60$) flow

From the frequency associated to the lift coefficient's oscillation (f) we can define the Strouhal number (St) as:

$$St = \frac{f \cdot D}{u_{max}} \quad (4.29)$$

Figure 4.35 shows the good agreement existing between the Strouhal number calculated with the FVM and the one obtained with the LBM. Small differences can be attributed to the lack of mesh refinement close to the obstacle and to the fact that the FVM simulation is 3D, whilst the LBM simulation is limited to two dimensions.

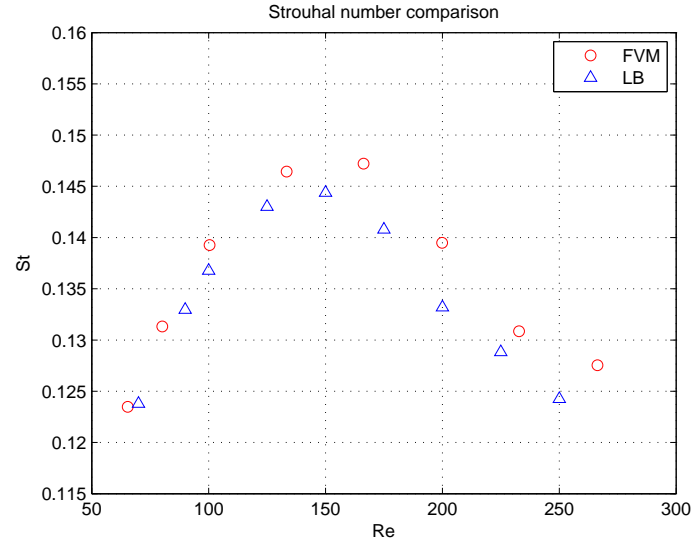


Figure 4.35 Strouhal numbers at different Re

CHAPTER 5 THREE-DIMENSIONAL LBM VERIFICATION AND VALIDATION

The aim of this chapter is the verification and validation of a 3D implementation of the LBM. At first the 3D LBM will be described. Then, the code for a pressure driven flow between two flat infinite plates will be validated. For this case, since two of the three dimensions are considered as infinite, we can validate the results by comparison with the well know 2D Poiseuille profile. Then, we will shift our attention to the pressure driven flow in a duct. For this case, an approximate analytical solution exists in literature. The procedure to obtain it will also be outlined. Finally, turbulence modeling will be investigated in combination with the LBM. In particular, a Large Eddy Simulation of a turbulent square jet in a cavity will be performed and validated by comparison to the experimental work of [35].

5.1 Description of Three Dimensional LBM

In 3D, several lattices with different characteristics exist. The most common one in literature is the three dimensional, nineteen velocities (D3Q19) cubic lattice. Other lattices are the D3Q15 (3D with 15 velocities) and the D3Q27 (3D with 27 velocities). According to the literature, the D3Q19 lattice grants a good trade-off between sufficient precision and competitive computation cost [12] and was therefore chosen for the present work. The three lattices performances have been compared in Appendix A.

Both the BGK-SRT and MRT collision models exist in 3D, and are implemented via the same equations presented for the 2D, respectively Equations 2.6 and 3.22. What changes, for the BGK-SRT, are the velocity discretization characteristics and associated weights. For the MRT collision, new expressions need to be found for the transformation matrix M and for the equilibrium distributions in the moment space. The lattice length units and time step are, just like the 2D case considered as unitary.

In the generic cubic lattice, shown in Figure 5.1, we can identify all the cells associated to all the available discretization directions for the velocity, labeled by squares. These can be grouped according to their velocities in lattice units:

- Area 0: the central cell has null velocity.
- Area 1: the six cells closer to the central have a speed of 1 lu/ts
- Area 2: the twelve cells located on the midpoints of the sides have a speed of $\sqrt{2}$ lu/ts

- Area 3: the eight cells located on the vertexes have a speed of $\sqrt{3}$ lu/ts

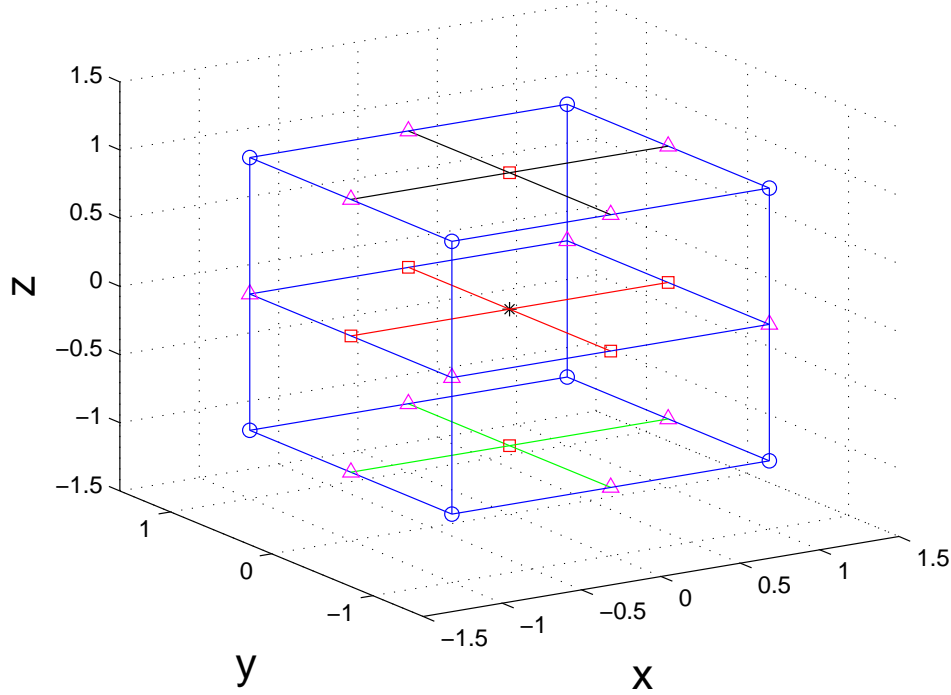


Figure 5.1 Generic cubic lattice unit. Area 0 cells are black and star-shaped, Area 1 cells are red squares, Area 2 cells are violet triangles and Area 3 cells are blue circles

This grouping enables us to quickly describe the D3Q15, D3Q19 and D3Q27 lattices. The D3Q15 lattice will have cells found in Areas 0, 1 and 3, the D3Q19 lattice will have cells in Areas 0, 1 and 2 and the D3Q27 will include cells belonging to all Areas.

The D3Q19 lattice is shown in Figure 5.2. The associated discrete velocities are:

$$e_i = \begin{cases} (0, 0, 0) & i = 1 \\ (\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1) & i = 2, 3, \dots, 7 \\ (\pm 1, \pm 1, 0), (\pm 1, 0, \pm 1), (0, \pm 1, \pm 1) & i = 8, 9, \dots, 19 \end{cases} \quad (5.1)$$

and the corresponding weight coefficients are:

$$w_i \begin{cases} 1/3 & i = 1 \\ 1/18 & i = 2, 3, \dots, 7 \\ 1/36 & i = 8, 9, \dots, 19 \end{cases} \quad (5.2)$$

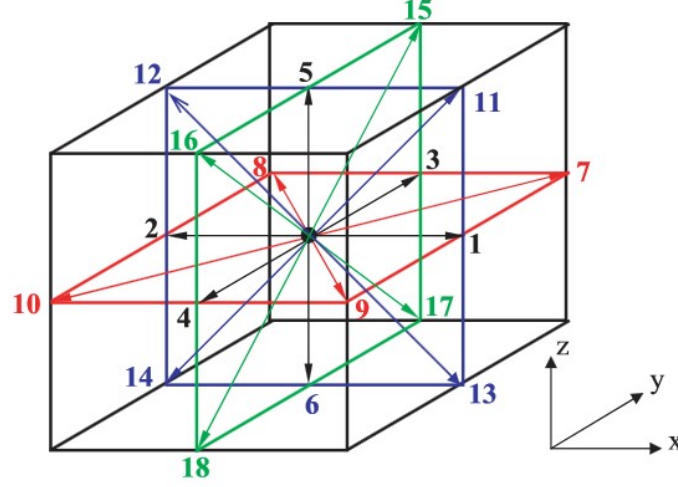


Figure 5.2 D3Q19 cubic lattice from [45]

Just like in 2D the density and momentum are recovered via sums of the distribution functions:

$$\begin{cases} \rho = \sum_{i=1}^{19} f_i \\ \rho \mathbf{u} = \sum_{i=1}^{19} \mathbf{e}_i f_i \end{cases} \quad (5.3)$$

5.1.1 Three-Dimensional Multiple Relaxation Times collision

Given that the expression of the collision process for the MRT collision remains the same as the 2D implementation (Equation 3.22), we need to define a new transformation matrix for the transition between the velocity and moment spaces. Because of the 19 velocities compatible with this lattice it will have 19 rows and 19 columns:

$$M = (\varphi_1, \varphi_2, \varphi_3, \dots, \varphi_{17}, \varphi_{18}, \varphi_{19}) \quad (5.4)$$

The coefficients of the M matrix are calculated following expressions given in Appendix B.

We also need to find an expression for the equilibrium moments which are related to different parameters such as density, energy (e), energy square (ϵ), momenta (j_x, j_y, j_z), energy flux (q_x, q_y, q_z) and the symmetric stress tensor ($p_{xx}, p_{yy}, p_{zz}, p_{xy}, p_{yz}, p_{xz}$). Their expressions are given by:

$$\left\{ \begin{array}{l} m_1^{eq} = \rho \\ m_2^{eq} = e = -11\delta\rho + \frac{19}{\rho_0} \mathbf{j} \cdot \mathbf{j} \\ m_3^{eq} = \epsilon = \omega_e \rho + (\omega_{ej}/\rho_0)(j_x^2 + j_y^2 + j_z^2) \\ m_4^{eq} = j_x \\ m_5^{eq} = q_x = -2/3 j_x \\ m_6^{eq} = j_y \\ m_7^{eq} = q_y = -2/3 j_y \\ m_8^{eq} = j_z \\ m_9^{eq} = q_z = -2/3 j_z \\ m_{10}^{eq} = p_{xx} = (1/\rho_0)(2j_x^2 - (j_y^2 + j_z^2)) \\ m_{11}^{eq} = \pi_{xx} = \omega_{xx} p_{xx} = \omega_{xx}(1/\rho_0)(2j_x^2 - (j_y^2 + j_z^2)) \\ m_{12}^{eq} = p_{yy} = 1/\rho_0(j_y^2 - j_z^2) \\ m_{13}^{eq} = \pi_{yy} = \omega_{yy} p_{yy} = \omega_{yy}/\rho_0(j_y^2 - j_z^2) \\ m_{14}^{eq} = p_{xy} = j_x j_y / \rho_0 \\ m_{15}^{eq} = p_{yz} = j_y j_z / \rho_0 \\ m_{16}^{eq} = p_{xz} = j_x j_z / \rho_0 \\ m_{17}^{eq} = m_{x,y,z} = m_{18}^{eq} = m_{19}^{eq} = 0 \end{array} \right. \quad (5.5)$$

The parameters ω_e , ω_{xx} and ω_{ej} are chosen to optimize the stability of the model [12]. Typically the following values are used:

$$\omega_e = 0 \qquad \omega_{xx} = 0 \qquad \omega_{ej} = -475/63$$

For the collision matrix \hat{S} , the same considerations presented for the 2D MRT collision are valid. It will then also be a diagonal matrix containing the relaxation coefficients associated to each of the moments:

$$\hat{S} = \text{diag}(s_1, s_2, s_3, \dots, s_{17}, s_{18}, s_{19}) \quad (5.6)$$

The values for the relaxation coefficients are chosen to optimize stability. Their most common values available in literature are $s_2=1.19$, $s_3=s_{11}=s_{13}=1.4$, $s_5=s_7=s_9=1.2$ and $s_{17}=s_{18}=s_{19}=1.98$ [24]. The relaxation coefficients associated to the equilibrium moments related to the stress tensor, depend on the viscosity:

$$s_{10} = s_{12} = s_{14} = s_{15} = s_{16} = \frac{2}{1 + 6\nu} \quad (5.7)$$

The other relaxation coefficients, s_1 , s_4 , s_6 and s_8 can be set as null.

The \hat{S} matrix optimized for stability will then be the following:

$$\hat{S} = \text{diag}(0, s_2, s_3, 0, s_5, 0, s_5, 0, s_5, s_{10}, s_3, s_{10}, s_3, s_{10}, s_{10}, s_{10}, s_{17}, s_{17}, s_{17}) \quad (5.8)$$

5.2 Pressure Driven Flow

The first validations which were performed to test the 3D implementation of the developed LBM code were the pressure driven flow between two infinite parallel plates and the pressure driven channel flow. For the first case, due to the fact that the plates are infinite in the streamwise and longitudinal directions, the steady state solution is independent on both these directions. The obtained solution can therefore be compared to the well-known 2D Poiseuille profile.

The pressure driven flow in a square channel is more complex, since the fluid interacts with walls both in the longitudinal and transverse directions. LBM results will be compared with the quasi-analytical solution obtained by [42].

5.2.1 Validation 1: Flow between infinite, parallel plates

The first performed validation test case for the 3D LBM code is the flow between two infinite, parallel plates. The geometry is shown in Figure 5.3.

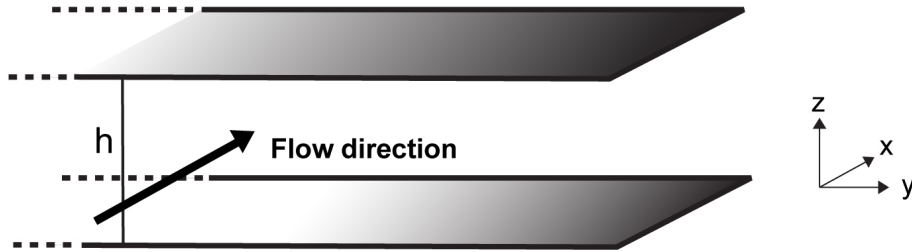


Figure 5.3 Geometry configuration

The domain is infinite both in the streamwise and longitudinal directions. We then impose periodic boundary conditions on the flow entry and exit sections and on the right and left borders. On the upper and lower boundaries a full way bounce-back condition was implemented, in the same way as the 2D case (Equation 3.13). The vector containing the "opposite" directions for the D3Q19 molecule is:

$$\begin{cases} \text{directions} &= [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] \\ \text{opposite}_{\text{directions}} &= [1, 3, 2, 5, 4, 7, 6, 11, 10, 9, 8, 15, 14, 13, 12, 19, 18, 17, 16] \end{cases} \quad (5.9)$$

The pressure gradient is modeled as a body force at the microscopic level and implemented in the same way as the 2D LBM.

The convergence criterion is related to the L^2 -norm as indicated in Section 4.2. Steady-state is considered as reached when the difference of the L^2 -norm of the error between successive 2000 iterations (relative error) is less than 10^{-7} .

Simulations are run both with BGK-SRT and MRT collisions for a Reynolds number of 500. The Reynolds number depends on the distance between the plates, the average speed of the Poiseuille profile and the fluid viscosity. In the LBM simulation, the average speed is set to be $u_{ave}=0.05$ lu/ts. The imposed distance between the plates is set to be of 60 lattice units. The viscosity is chosen according to the other parameters, in order to respect the imposed Reynolds number.

Figure 5.4 shows the excellent agreement found between the analytical Poiseuille profile and the velocity obtained with the LBM simulation.

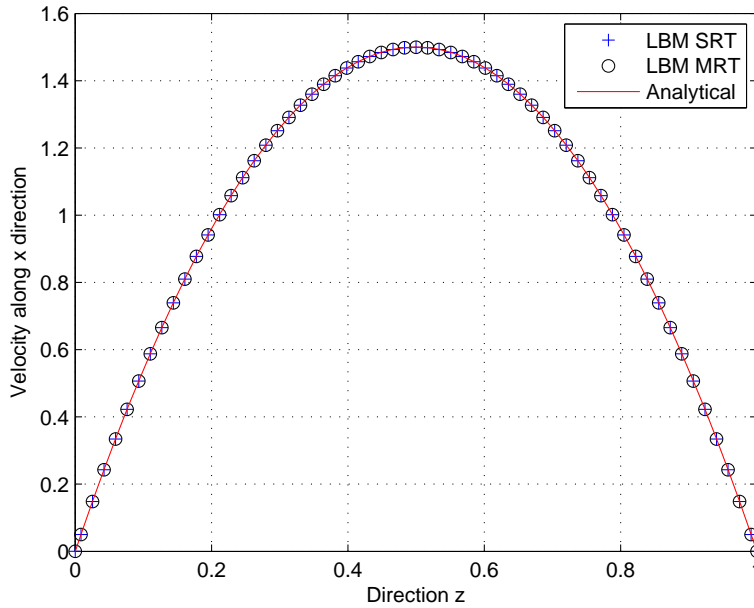


Figure 5.4 Flow between parallel plates: streamwise velocity profile

The transverse direction is rendered nondimensional according to the node number on that direction and the velocity is normalized according to the average speed of the Poiseuille profile. As expected, given considerations in Section 4.2, we notice that the maximum velocity reached is equal to 1.5 times the average speed. Both BGK-SRT and MRT collisions are plotted on the same graph, and match correctly the analytical solution.

Figure 5.5 presents the absolute error and convergence for increasing iterations. The absolute error indicates the L^2 -norm of the error between the LBM results and the analytical solution. The convergence, on the other hand, indicates the difference between successive 2000 iterations calculations of the L^2 -norm of the error. We can clearly identify that the final iteration takes place when the convergence criterion is met. An indication on the time duration of the developed code for this and all other test cases is provided in Appendix D.

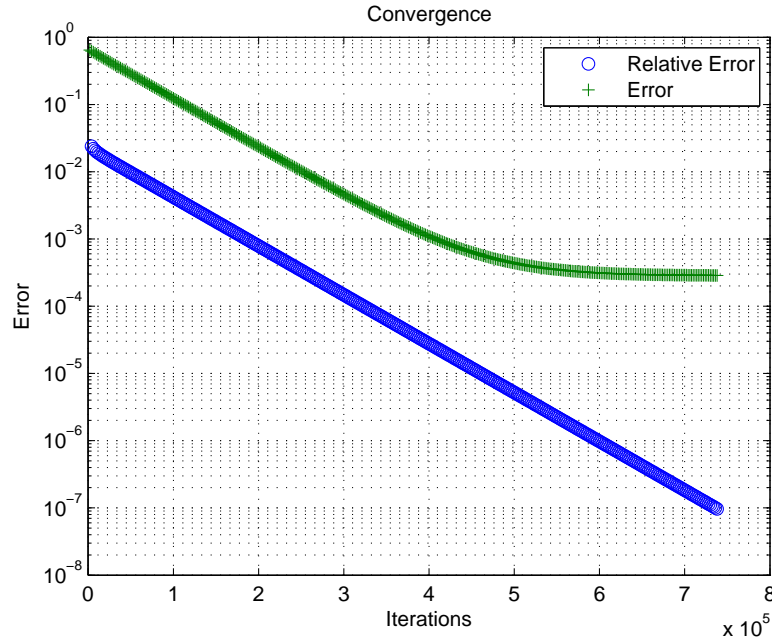


Figure 5.5 Flow between parallel plates: relative and absolute error trends

In order to validate the LBM code in all of the three directions, the flow streams in the y and z directions were also simulated. The obtained velocity profiles had perfect correspondence to the analytical solution in both cases.

5.2.2 Validation 2: Pressure Driven Channel flow

The second validation performed is the simulation of a pressure driven square channel flow. Because of the interactions between the stream and the walls both in the longitudinal and transverse directions, the flow is more complex than the previous case. The geometry is shown in Figure 5.6.

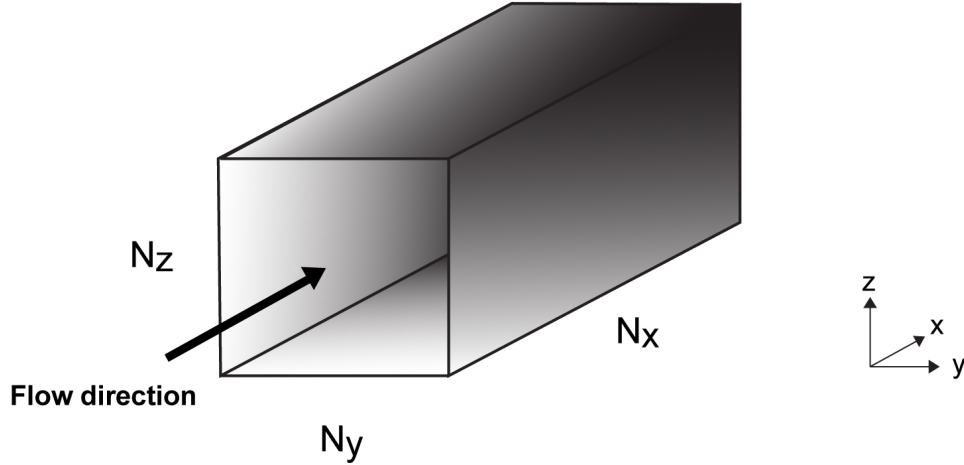


Figure 5.6 Geometry configuration

In order to validate the results obtained with the developed LBM code, an analytical expression for the velocity profile has to be found. [42] were able to obtain an analytical velocity profile, given the approximation of an almost square duct, for a laminar pressure-driven channel flow. Their procedure is detailed in Appendix C.

Given the geometry presented in Figure 5.6, it is evident that since we are looking for a steady state solution we can impose a periodic boundary condition between the inlet and exit sections. On the walls, on the other hand, a full-way bounce-back boundary condition is imposed. The Reynolds number is set to 500, well below the laminar-turbulent transition limit for channel flows (Re of around 2000). Its computation is based on the channel's height, the reference speed in lattice units (set to be 0.05 lu/ts to minimize compressibility effects) and the fluid's viscosity.

The pressure gradient was included using the same procedure presented in Section 4.2. The

convergence criterion is based on the L^2 -norm described in Section 4.2. Steady state is considered as reached when successive 2000 iterations' errors differ of less than 10^{-9} . The L^2 -norm of the error between the LBM and the quasi-analytical solution is shown in Figure 5.7. In this same Figure we can also observe the difference between the L^2 -norm calculations between successive 2000 iterations, which is used to measure the solutions convergence. We therefore observe that, when the convergence reaches 10^{-9} , the code is stopped, since the steady state is considered as reached.

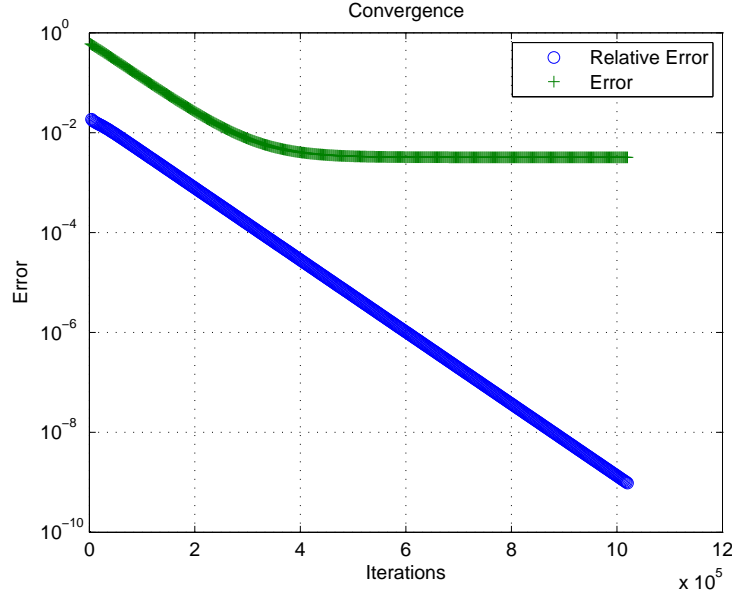


Figure 5.7 Square Duct flow: absolute and relative error trend

The results presented in Figures 5.8 and 5.9 are in terms of the streamwise velocity for a BGK-SRT collision model and a mesh of $3 \times 121 \times 121$. The plotted profile in Figure 5.8 is located at the channel's half length and mid-height (normal to the streamwise direction). The longitudinal direction is normalized according to its length in lattice units and the speed is rendered nondimensional with respect to the reference speed. Because the solution is steady state and due to the periodicity in the streamwise direction, the mesh density in the streamwise direction does not affect the result's accuracy.

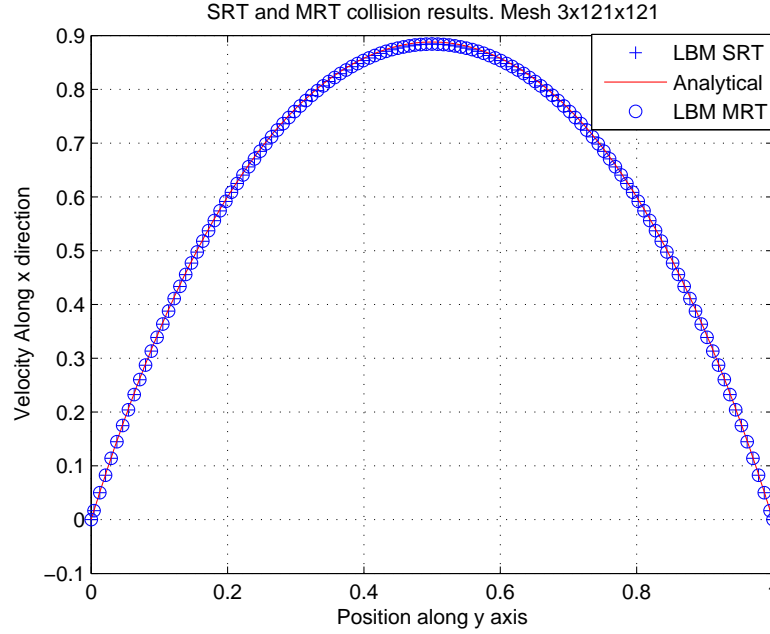


Figure 5.8 Comparison between quasi-analytical and LBM velocity profile

Despite the fact that the calculated analytical solution is based on a quasi-square section and the LBM results are related to a perfectly squared channel, we observe good agreement between the results.

Figure 5.9 gives a better idea of the actual 3D nature of the flow since it is a top view of the paraboloid representing the 3D expression of the streamwise speed. Results are plotted in terms of the normalized longitudinal and transverse directions, for a section located at mid channel's length (normal to the streamwise direction).

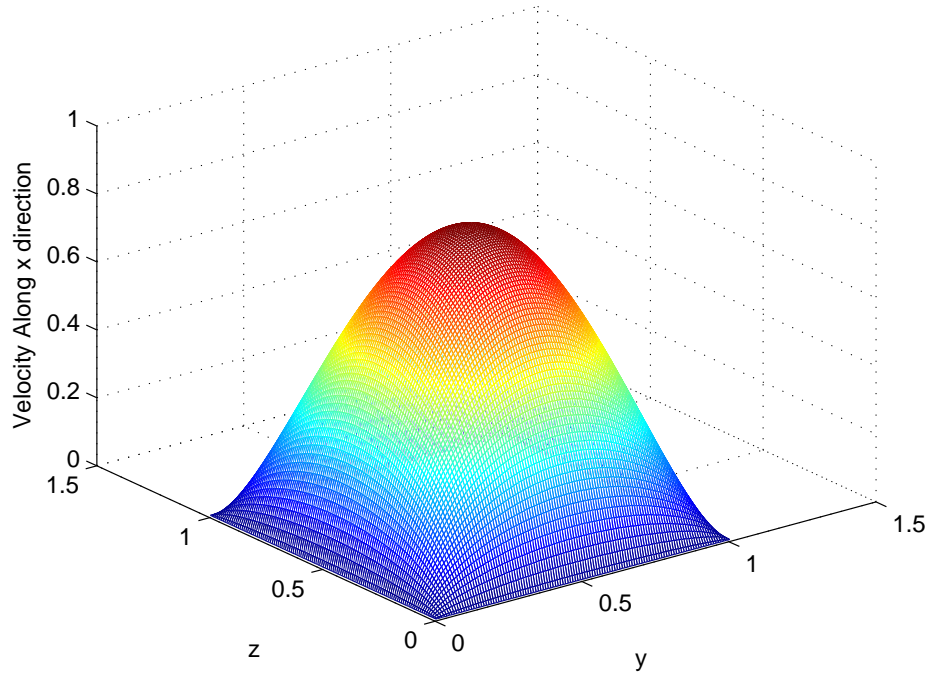


Figure 5.9 Square Duct flow: 3D plot of the streamwise speed

We can now verify the convergence order of the 3D LBM code together with its *GCI* for infinite mesh and the errors associated to the discretization.

Given the three incrementally fine meshes presented in Table 5.1 we obtain the convergence and error parameters presented in Table 5.2 via the procedure presented in Section 4.1. Results are obtained for $Re=500$ and lattice reference speed of 0.05 lu/ts. It should be noted that the only nodes influencing the results accuracy are the ones discretizing the channel's section. The examined parameter of the simulation, ϕ , is the streamwise velocity at mid channel's length and height.

Table 5.1 Square duct flow: grid characteristics

Mesh Type	Node Number	Parameter r
3. Coarse	31x31	
2. Medium	61x61	$r_{21}=2.00$
1. Fine	121x121	$r_{21}=2.00$

Table 5.2 Square duct flow: convergence parameters

Collision	Boundary Condition	p	e_a^{21}	e_{ext}^{21}	GCI
SRT	Bounce-back	0.981	1.81%	1.65%	2.19%
MRT	Bounce-back	0.984	1.72%	1.71%	2.27%

We observe that the 3D convergence order of the LBM is around 1, given that the bounce-back conditions are of order 1 in terms of accuracy. The error parameters and the GCI are low, which implies that the selected meshes are fine enough to correctly capture the flow's characteristics. In particular having a low GCI is important, because this indicates that there isn't too much difference between results obtained with the fine mesh and the ones obtained via an extrapolated, infinitely fine mesh.

5.3 Square Turbulent Jet

The aim of this last section is the simulation of a turbulent flow with the LBM method. Turbulence modeling with the LBM is, at present, in its infancy phase, since not many simulations of turbulent flows are available in literature compared to the classic N-S based solvers. In fact several challenges in the LBM do exist when we have to model turbulence. Aside from the way we implement the turbulence model, which has to take into account that simulations are run in a "lattice based" domain, we should also consider that the LBM is sometimes unstable at high Reynolds numbers. As an example, it has been proven, in Section 4.4, that, when the Reynolds number is too high, the most widespread collision implementation, BGK-SRT, is unstable. Other issues exist related to the boundary conditions and the correct modeling of flow-wall interactions [20].

In this context, it was judged useful to try and implement a turbulence model in the LBM. The pioneering work of [45], who implemented a simulation of the square turbulent jet in a cavity, is a good starting point and valid reference to try and address the "turbulence challenge" in the LBM.

The LBM simulation of a turbulent square jet in a square cavity (geometry is reported in Figure 5.12) will be implemented using the Large Eddy Simulation turbulence model. Results will be compared to the experimental work of [35].

First of all, however, the LES turbulence model with Smagorinsky sub-grid scale model will be briefly introduced and its implementation in the LBM will be outlined.

5.3.1 Turbulence in the LBM

Turbulence is a very complex phenomenon, extensively studied and modeled over the last decades. It concerns a large number of applications, including fuel-air mixing in internal combustion engines, aircraft design, weather predictions and, in general, it should be said that many of the fluid dynamic simulations of real world situations need turbulence modeling. The term turbulence probably finds its origin in the work of Leonardo da Vinci, scientist and inventor of the 16th century, who first represented the swirling movement of water in one of his sketches, presented in Figure 5.10. Even at the time, turbulence was interesting in particular because people were exploring possible energy or transport applications of water flow in rivers. The study of turbulence became an actual experimental science thanks to the research by Osborne Reynolds, responsible for the study of the laminar-turbulent transition in a fluid and also for the definition of the Reynolds number, still in use today as an indicator of the turbulence intensity.



Figure 5.10 Turbulence sketch by da Vinci

The main objective of the study of turbulence is being able to predict its effects, in order to control them. Turbulence has indeed both positive and negative consequences depending on the circumstances. On airplane wings, as an example, it increases drag but also delays the flow's detachment from the wing.

Turbulence can be described as a mostly three-dimensional chaotic process involving the following characteristics:

- Randomness and irregularity. Turbulent flows are deeply irregular and involve sudden and continuous variations of properties such as velocity and pressure. This is the reason why they are normally treated statistically. This is clearly shown by looking at the time history of the centerline velocity for the axial component of velocity in a turbulent jet, shown in Figure 5.11

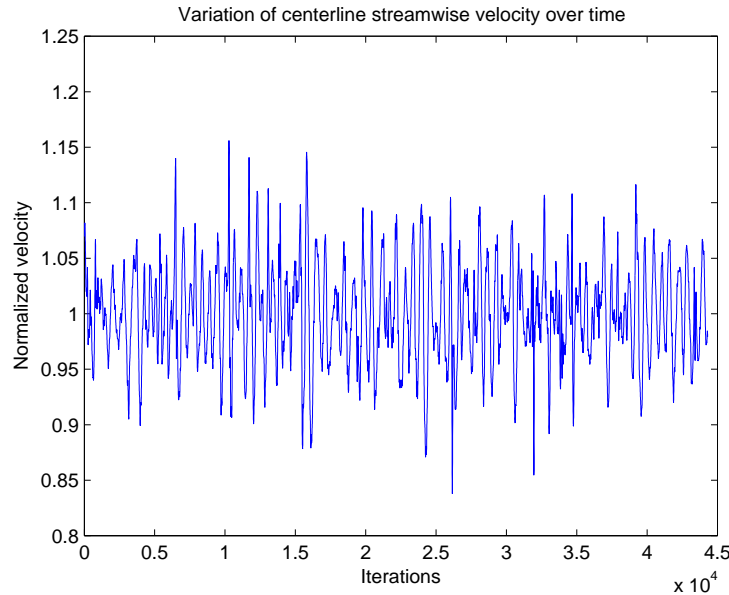


Figure 5.11 LBM simulation of turbulent jet at $Re=184000$. Normalized centerline velocity at a certain position along the channel's length

- Dissipation. Turbulence is not a self-sustaining process, and in order for it to occur energy must be constantly provided. In fact the viscous forces dissipate the kinetic energy in the flow via an increase in the internal energy of the flow.
- High Reynolds number. Turbulent flows always occur at large Reynolds numbers meaning they are characterized by the prevalence of inertia forces compared to viscous ones.
- Diffusivity. The supply of energy in turbulent flows tends to accelerate the mixing of fluid thus increasing mass, momentum and energy transports.
- Three-dimensional vorticity. Turbulent flows are characterized by the formation of vortices which are inherently a three-dimensional phenomenon.

Turbulence causes the formation of localized flow structures named "eddies" of different length scales: most of the energy is contained in the larger ones, while the smaller scales are respon-

sible for the viscous dissipation of energy. In fact Kolmogorov's universal equilibrium theory states the existence of an energy cascade from the larger high energy eddies to the smaller ones. He suggested that large eddies are virtually unaffected by viscosity and transfer their energy to the small ones, with a rate equal to their rate of kinetic energy dissipation into heat. Generally the ratio of the Kolmogorov length scale to integral scale, which are respectively the smallest and largest scales, is in the order of $Re^{-3/4}$. This means that if we had to define a 3D mesh capable of correctly capturing the properties of the smallest scales, we would need in the order of $Re^{9/4}$ nodes. Given that turbulent applications are characterized by very high Reynolds numbers, we understand that this approach is not ideal.

Kolmogorov suggested the hypothesis that the smaller scales of motions are isotropic, since the "directional information" of the larger eddies is lost in the scale reduction process. This implies that they can be treated under a statistical point of view.

Various methods exist for the study of turbulence such as the DNS (Direct Numerical Simulation), the LES (large eddy simulation) and the RANS (Reynolds averaged Navier Stokes), each with their own advantages and limitations.

A simulation where all of the length scales are solved is called a Direct Numerical Simulation. Due to its extremely high computational cost it was not viable until the 1970s when powerful enough computers became available. In fact, most applications involve too many scales to be directly solved, and therefore DNS applicability is limited to low Reynolds numbers, which involve more limited scales of motion.

The Reynolds-averaged Navier–Stokes equations (RANS) are another option for the simulation of turbulent flows. The idea behind this approach is the averaging of the N-S equations which are then solved in terms of average quantities only. This method is characterized by a relatively low computational cost which comes with a generally lower precision.

Although DNS has many advantages thanks to its accuracy and straightforward implementation, its requirements in terms of computation cost are most often too high. On the other hand, the RANS are far less computationally intensive, but also offer less accurate results.

An intermediate option for turbulence modeling are Large Eddy Simulations. This method provides a good trade off between the two others, since it is less costly in terms of computation time with respect to the DNS and grants more accuracy than the RANS approach. A Large Eddy Simulation (LES) involves the application of a filtering length representative of the largest scales of the turbulent motion. Scales smaller than the filtering length are not directly solved and their influence is merely modeled.

More specifically, the filtering operation decomposes the velocity into a resolved component

and a residual sub grid scale (SGS) component. The residual unsolved motions' effect is included in a residual stress tensor, τ_{ij}^r . The simplest model used to solve this tensor was proposed by [37]. First of all, the "turbulent" or eddy viscosity ν_t is used to relate the residual stresses to the strain-rate tensor:

$$\tau_{ij}^r = -2\nu_t \bar{S}_{ij} \quad (5.10)$$

The strain-rate tensor is built given the filtered velocity gradients $\partial \bar{u}_i / \partial x_j$ as:

$$\bar{S}_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \quad (5.11)$$

where variables presented with an overbar indicate filtered quantities. Then, the eddy viscosity is modeled as [32]:

$$\nu_t = (C_s \Delta_x)^2 \bar{S} \quad (5.12)$$

where C_s represents the Smagorinsky coefficient, Δ_x represents the filtering length and \bar{S} is the characteristic filtered rate of strain defined as:

$$\bar{S} = \left(2 \sum_{i,j} \bar{S}_{ij} \bar{S}_{ij} \right)^{0.5} \quad (5.13)$$

The Smagorinsky coefficient is not a universal constant which is a significant shortage of this turbulence model. It has to be determined according to the flow type we are modeling. A value of around 0.1 has been found to give the best results for a variety of flows and is the default value in some commercial CFD software [7]. On what concerns the filtering length, the usual approach for the LBM-LES simulation is setting it as equal to the unitary lattice length. In this way the lattice mesh is used to resolve the higher length scales of motion, while the turbulent viscosity account for what happens at length scales lower than the grid spacing [44, 18].

In the LBM, the viscosity resulting from the modeling of the sub-grid scales, named turbulent viscosity, is added to the simulated fluid's original viscosity, ν_0 . In the BGK-SRT collision implementation, the new relaxation parameter τ_{tot} should now take into account the additional viscosity [44]:

$$\tau_{tot} = \tau_0 + \tau_t = 3\nu_{tot} + 0.5 = 3(\nu_0 + \nu_t) + 0.5 \quad (5.14)$$

where ν_0 and τ_0 indicate the "original" (without turbulence model) kinematic viscosity and relaxation time.

The turbulent viscosity is related to the strain-rate tensor, as shown in Equation 5.12. The strain rate tensor can in turn be obtained from the filtered momentum flux \bar{Q}_{ij} by [44]:

$$\bar{S}_{ij} = \frac{1}{2\rho_0 C_s^2 \tau_{tot}} \bar{Q}_{ij} \quad (5.15)$$

The filtered momentum flux is in turn defined from the non-equilibrium distribution functions:

$$\bar{Q}_{ij} = \sum_{\alpha} e_{\alpha,i} e_{\alpha,j} (\bar{f}_{\alpha} - \bar{f}_{\alpha}^{(eq)}) \quad (5.16)$$

where α represent the velocity directions compatible with the lattice. It should be noted that the computation of the filtered momentum flux, and consequently of the turbulence viscosity, is local to each node.

If we consider Equations 5.12 to 5.16 we notice they form a system of five equations with five unknowns (ν_t , τ_t , Q_{ij} , S_{ij} and \bar{S}). Once the system is solved, the following expression is obtained for the turbulence related part of the relaxation coefficient:

$$\tau_t = \frac{1}{2} \left(\sqrt{\tau_0^2 + \frac{2 (C_{sm} \Delta_x)^2 \bar{Q}}{\rho_0 C_s^4}} - \tau_0 \right) \quad (5.17)$$

We can see that the value of τ_t should be locally adjusted, according to the local magnitude of the filtered momentum flux.

For the LBM implemented using the MRT collision, the strain rate tensor is dependent on the non-equilibrium moments. These have the following expressions, reported by [44]:

$$\left\{ \begin{array}{l} m_1^{neq} = \frac{38}{3} (j_x + j_y + j_z) \\ m_9^{neq} = -\frac{2}{3} (2j_x - j_y - j_z) \\ m_{11}^{neq} = -\frac{2}{3} (j_y - j_z) \\ m_{13}^{neq} = -\frac{1}{3} (j_x + j_y) \\ m_{14}^{neq} = -\frac{1}{3} (j_z + j_y) \\ m_{15}^{neq} = -\frac{1}{3} (j_x + j_z) \end{array} \right. \quad (5.18)$$

From which the components of the symmetric strain rate tensor can be calculated via the following expressions:

$$\left\{ \begin{array}{l} S_{xx} = -\frac{m_1^{neq}}{38\rho_0} - \frac{m_9^{neq}}{2\rho_0} \\ S_{yy} = -\frac{m_1^{neq}}{38\rho_0} + \frac{m_9^{neq}}{4\rho_0} - \frac{3m_{11}^{neq}}{4\rho_0} \\ S_{zz} = -\frac{m_1^{neq}}{38\rho_0} + \frac{m_9^{neq}}{4\rho_0} + \frac{3m_{11}^{neq}}{4\rho_0} \\ S_{xy} = -\frac{3m_{13}^{neq}}{2\rho_0} \\ S_{yz} = -\frac{3m_{14}^{neq}}{2\rho_0} \\ S_{xz} = -\frac{3m_{15}^{neq}}{2\rho_0} \end{array} \right. \quad (5.19)$$

From these, the turbulent viscosity can be calculated using Equation 5.12.

The total viscosity is then used for the computation of the viscosity dependent relaxation coefficients of the \hat{S} matrix reported in Equation 5.8.

5.3.2 LBM simulation results

In the experimental work by [35] a jet of air exits a square slot of side $h_{slot} = 4cm$ and enters a square cavity of size $244^2 \times 366$ cm³. The mean streamwise velocity at the entry slot is $u_{slot} = 60m/s$. The Reynolds number for this configuration is based on the slot's width, the viscosity of air and the mean speed:

$$Re = \frac{h_{slot} u_{slot}}{\nu_{air}} \quad (5.20)$$

For the tested conditions the Reynolds number is around 184000.

The LBM implementation aims to study the jet characteristics in an area close to the jet. The geometry of the LBM simulation is presented in Figure 5.12.

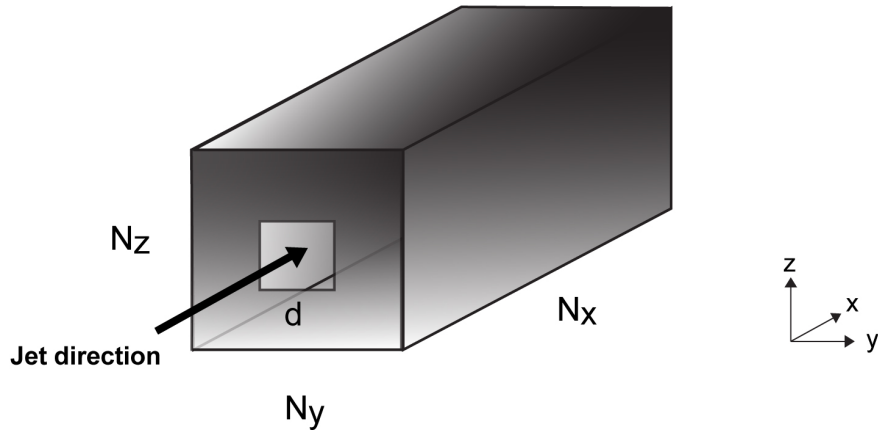


Figure 5.12 Geometry configuration

On the x streamwise direction $N_x = 600$ nodes are used. On the y spanwise direction and on the z vertical direction $N_y = N_z = 121$ nodes are used. The slot opening width, d , is set as equal to one fifth of the spanwise direction. The slot is located on the $x = 0$ plane with its center located exactly at mid height and width of the cavity. An equivalent diameter D_e is defined as equal to the diameter of a circle which has the same area as the slot opening:

$$A_{slot} = d^2 = A_{equi,circle} = \frac{\pi D_e^2}{4} \quad \rightarrow \quad D_e = \frac{2d}{\sqrt{\pi}} \quad (5.21)$$

A null velocity condition is imposed on the $x = 0$ plane. This solution yields similar, but slightly more accurate, results with respect to the one obtained with bounce-back boundary conditions. At the walls, located in the vertical and spanwise directions, we also impose a null velocity boundary condition. The distributions in these positions are therefore equal to the equilibrium distributions calculated for null velocity.

At the outflow, a free boundary condition is applied. If we suppose that the flow close to the streamwise outlet boundary is fully developed, we can impose that the distributions at the outflow are equal to those at the nodes located one lattice unit before the outflow. We are therefore imposing the properties' derivatives as null at the outflow. This condition, although very effective in the study of laminar flows, is quite approximate for turbulent flows where the flow is always space and time dependent. The solution is therefore non physical in the proximity of the outflow. We should then focus our attention on the results close to the jet inlet only, since they are less influenced by the approximation introduced at the outlet.

Given that no specific information is available from the experimental setup of the turbulent jet in terms of the exact velocity profile at the inlet, we impose a uniform velocity profile. The density is also imposed as constant at the inlet.

$$u_{in,lb} = 0.1 \quad \text{lu/ts} \qquad \rho_{in,lb} = 1$$

We observe that the imposed velocity in lattice units keeps the Mach number of the simulation at $Ma = 0.17$, well below the limit after which the compressibility effects can no longer be neglected. The assumption of uniform velocity is an approximation in terms of the streamwise direction because it does not account for the boundary layer thickness. The spanwise and vertical velocities are set as null. This is also a quite strong approximation, since we can reasonably assume that the geometry behind the jet cause the spanwise and vertical velocity components to be non null at the jet exit.

Initially, the flow is at rest, except at the jet exit, with unitary density in all of the domain. From the equivalent diameter we can define the parameter T_0 to measure the simulation's duration:

$$T_0 = \frac{D_e}{u_{in}} \tag{5.22}$$

Sufficient iterations should be allowed to enable the jet's full development in the cavity. To obtain accurate results, the result's averaging period should also be chosen carefully. Starting

from the values found in [45], it was observed that simulations run for a "buffer" period of $70T_0$ and an averaging period of $100T_0$ yield good results. Longer "buffer" and averaging periods do not cause significant differences in the final results.

The centerline is defined as the line going from the jet entry slot to the outflow boundary located at mid-spanwise and vertical directions. The jet's half width is in turn defined as the distance from the centerline at which the velocity is equal to the half of the centerline's speed.

In the LBM-LES simulations the filtering length is most often set as equal to the lattice spacing, which in turn is often set as unitary [44]. In the present simulation: $\delta x = \delta t = \Delta_x = 1$, where δx and δt are the lattice spacing and time step (in lattice units) and Δ_x is the filtering length. This means that the large turbulent scales are resolved on the uniform Cartesian lattice. All the quantities involved in the LBM simulation (distributions, moments, velocities and densities) are therefore filtered quantities.

Both the BGK-SRT and MRT based collision operators were implemented. As shown in Figure 5.13 the implementation based on the BGK-SRT collision presents instabilities in the results, due to the high Reynolds number. This is consistent with the observations by [45]. The rest of the results will therefore be obtained with a LBM-LES implementation with the MRT collision operator.

First of all, we will look at the extent of the jet's penetration in the still cavity. Figure 5.14 shows the centerline velocity in the area close to the jet exit. The velocity is normalized according to the maximum one. In the experimental profile, the maximum velocity is not reached at the outflow, but in the *Vena Contracta* region located very close to the jet exit.

Figure 5.15 gives an idea of the averaged streamwise velocity on a plane found at mid-height of the cavity. We can clearly observe the extent of the jet's penetration inside the square cavity.

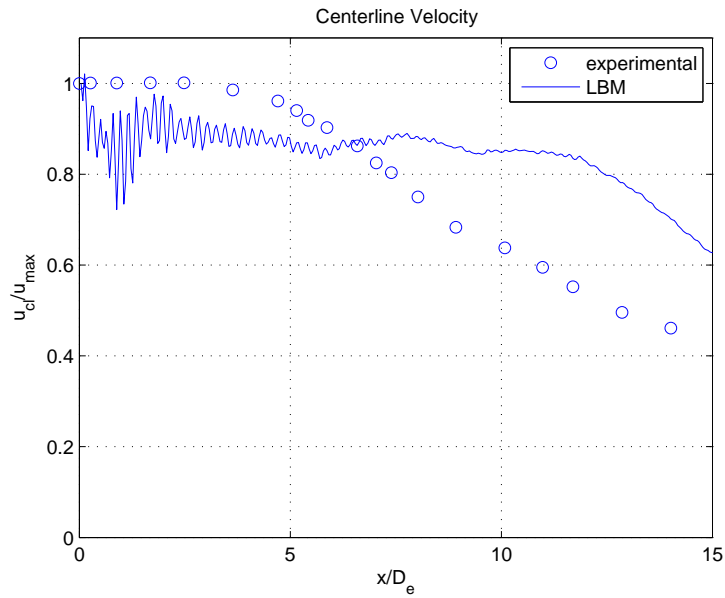


Figure 5.13 Square Turbulent Jet: streamwise normalized velocity at the centerline. BGK-SRT collision operator

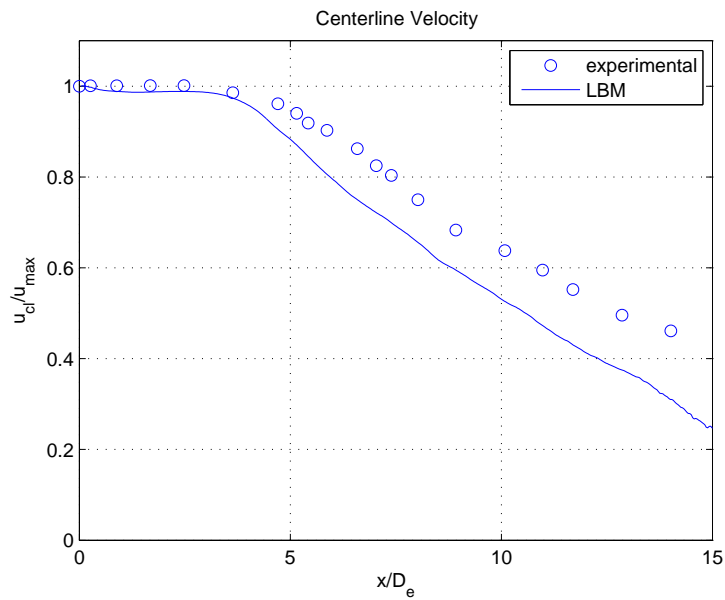


Figure 5.14 Square Turbulent Jet: streamwise normalized velocity at the centerline. MRT collision operator

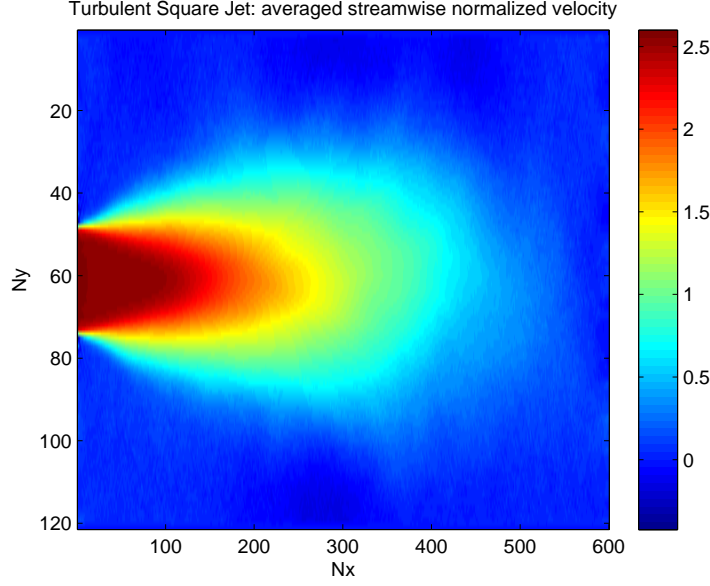


Figure 5.15 Square Turbulent Jet: averaged streamwise velocity in the domain (MRT collision operator).

The *Vena Contracta* effect doesn't appear to be captured by the LBM simulation. This can be explained by the imposed approximated inflow profile and by the lack of mesh refinement in the area close to the jet exit.

Figures 5.16 to 5.20 show the streamwise velocity profiles at four different locations on the channel's length:

$$\frac{x}{D_e} = 0.28 \qquad \frac{x}{D_e} = 1.121 \qquad \frac{x}{D_e} = 2.658 \qquad \frac{x}{D_e} = 7.088$$

Results are plotted at mid vertical height, along the spanwise direction. The experimental values are obtained from Figures in the [35] reference. The streamwise velocities are normalized according to the local centerline velocity. The spanwise direction is normalized according to the local jet's half width. It should be noted that the selected locations do not correspond exactly to a nodal position in the LBM simulation. A linear interpolation between the results obtained on the adjacent nodes in the streamwise direction is then performed.

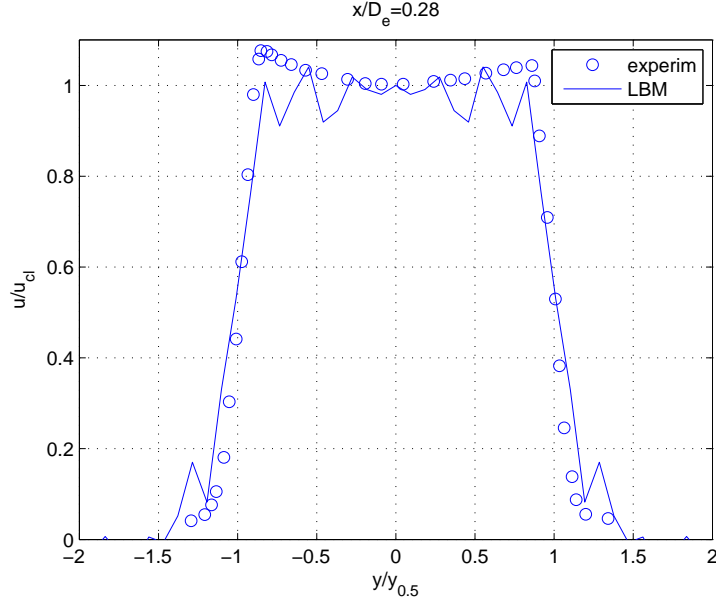


Figure 5.16 Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 0.28$. BGK-SRT collision operator

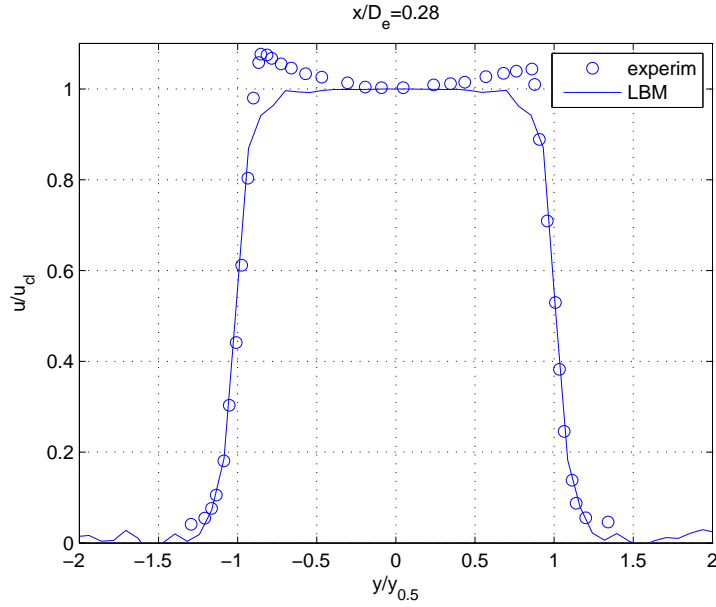


Figure 5.17 Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 0.28$. MRT collision operator

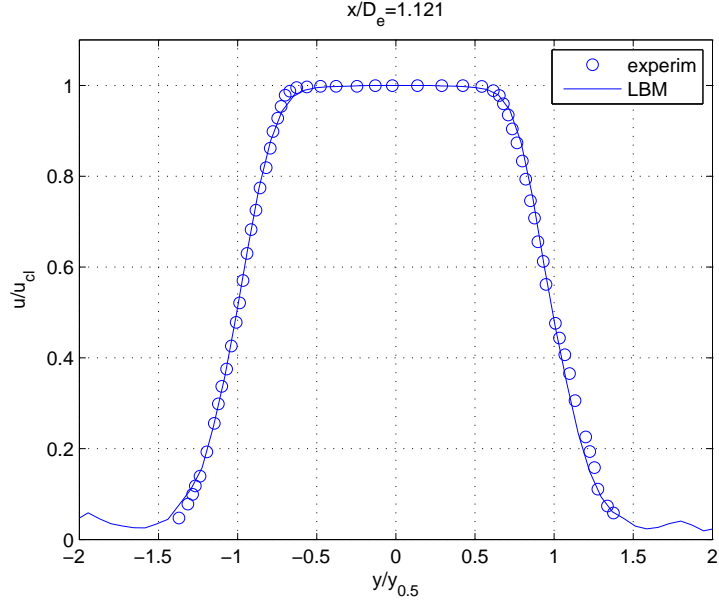


Figure 5.18 Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 1.121$. MRT collision operator

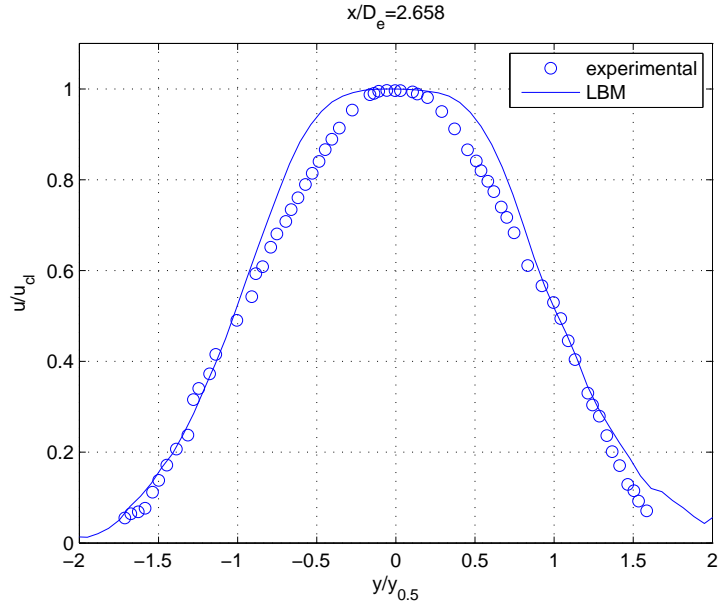


Figure 5.19 Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 2.658$. MRT collision operator

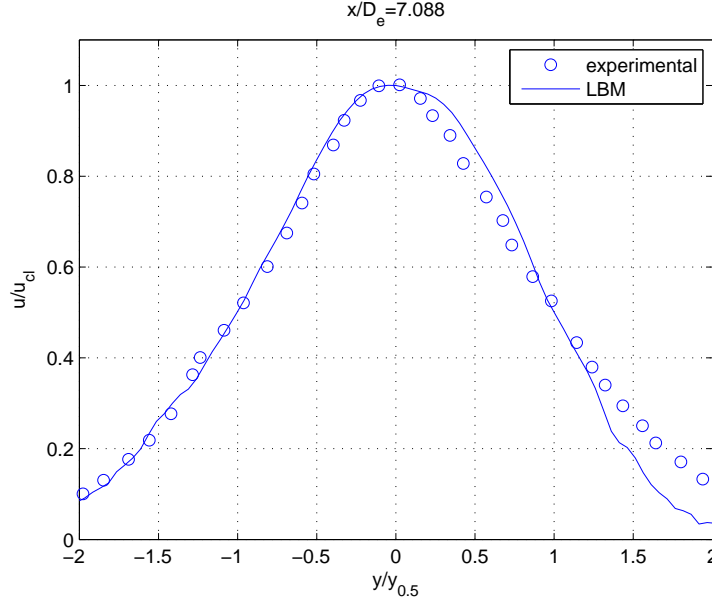


Figure 5.20 Square Turbulent Jet: streamwise normalized velocity at $\frac{x}{D_e} = 7.088$. MRT collision operator

We can see, in Figure 5.16, that the results obtained with the BGK-SRT collision are "noisy" and fluctuating due to the instability of the collision model. Figures 5.17 to 5.20 show the great agreement existing between experimental and LBM-MRT-LES results. We should consider that the boundary conditions and imposed inflow velocity profile introduce some approximation in the results. Some variables, such as the averaged streamwise velocity profiles will be less influenced because the approximation in terms of the streamwise component of inflow profile is quite small and because these are mean quantities, averaged over a consistent period. On what concerns the mean spanwise and vertical velocities, we understand that the strong approximation in the inlet profile prevents an exact comparison with experimental results.

Another interesting parameter to look at is the jet's half width. This parameter can be better measured in a LBM uniform Cartesian grid by linearly approximating the simulation's results in between nodes. This enables a more accurate measuring of the position at which the velocity is half of the centerline's one. Figure 5.21 shows excellent agreement, for the jet's half width, between the LBM simulation and the values obtained by [35]. We should note that, in the area close to the jet exit, measuring the half width is more approximated since the velocity gradients involved are high.

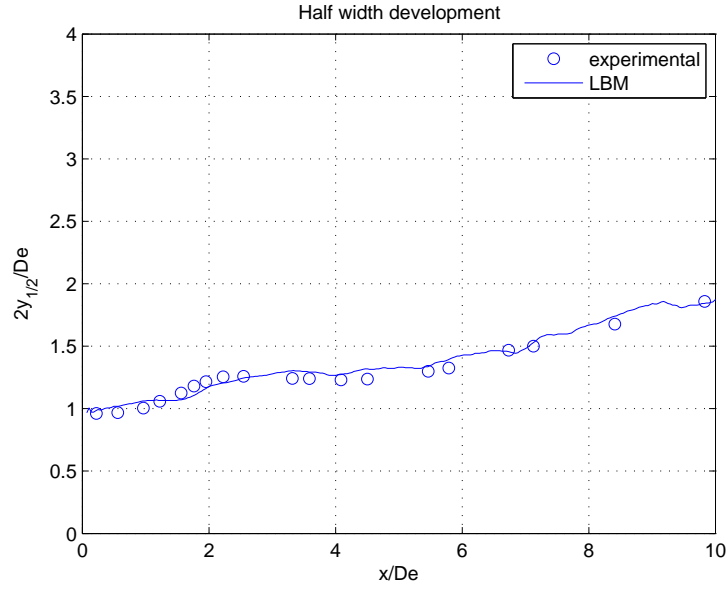


Figure 5.21 Square Turbulent Jet: jet half width

Non-circular jets are known to be characterized by the phenomenon of axis-switching [9]. This means that the jet will rotate around its axis as it penetrates the cavity. Figure 5.22 clearly shows that this phenomenon is captured by LBM simulations also. In fact we notice that, at a distance of $x = 3.6 \cdot d$ from the jet inlet, the streamwise velocity contour (circumference at which the velocity is half of the centerline velocity) has rotated of approximately 45° with respect to the jet exit.

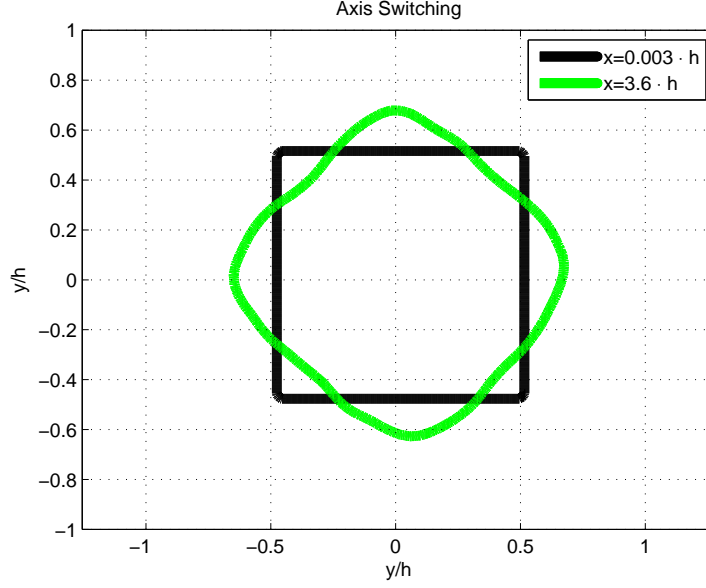


Figure 5.22 Half-velocity contour for the $x = 0.003 \cdot h$ and $x = 3.6 \cdot h$ sections

In the work of [35], we also find the experimental measurements of turbulence intensities which are used to give an idea of the "level of turbulence" in a flow. In fact, the mean quantities are not sufficient to describe turbulence since we do not measure the effect of the fluctuations around the mean values:

$$u_{total} = u_{mean} + u_{fluctuation} = u_{ave} + u' \quad (5.23)$$

Turbulence intensity calculations are based on the fluctuations only. In particular, they involve the calculation of their root mean square (rms) value. The rms of the fluctuations is defined as the square root of the time averaged fluctuations squared:

$$rms = \sqrt{\frac{1}{(t_{tot} - t_{av})} \sum_{i=t_{av}}^{t_{tot}} (u'(t_i))^2} \quad (5.24)$$

where t_{av} indicates the time instant at which the averaging of the results begins and t_{tot} indicates the final averaging time instant. In Figures 5.23 and Figure 5.24 we observe the normalized turbulence intensities for the streamwise and the spanwise velocities at different locations in the channel's length. As reported by [44], we know that these quantities are strongly influenced by the inlet profile and therefore we expect a merely qualitative resemblance between the experimental and LBM results. The inlet profile in the LBM simulation

is imposed as an uniform, streamwise only, velocity. It is then reasonable to assume that the approximation is stronger in terms of the spanwise imposed velocity (which is set as null) rather than in terms of the streamwise velocity. Results are then expected to be more accurate for the streamwise turbulence intensities with respect to the spanwise turbulence intensities, as proven by Figures 5.23 and 5.24.

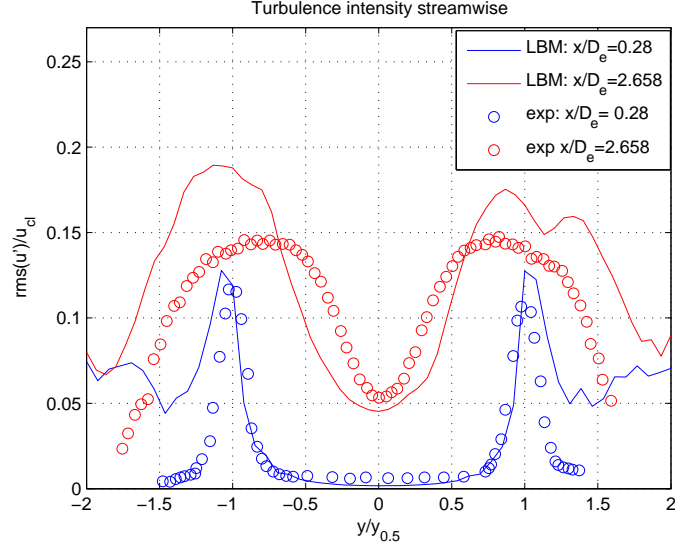


Figure 5.23 Square Turbulent Jet: Turbulence streamwise intensity at $\frac{x}{D_e} = 0.28$ and $\frac{x}{D_e} = 2.658$

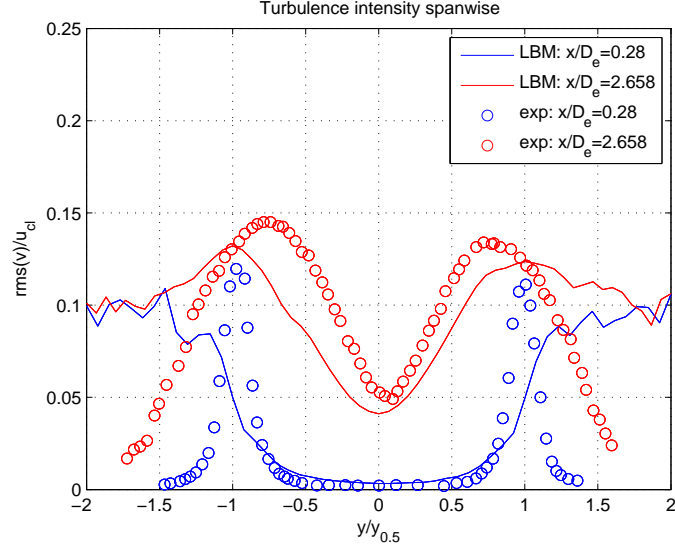


Figure 5.24 Square Turbulent Jet: Turbulence spanwise intensity at $\frac{x}{D_e} = 0.28$ and $\frac{x}{D_e} = 2.658$

We can see that the general trend of the profile is maintained even if there exists some difference between the local values. Differences appear to decrease as we move along the axial direction further away from the jet, since the influence of the inlet profile decreases.

In the sections close to the jet inlet, the turbulence intensity is greater at the interface between the jet flow and the cavity fluid, because of the high velocity gradients and local shear values. We notice that, in the LBM, the turbulence intensities do not fall to zero right at the jet's outer perimeter, but further away from it. In fact, the numerical error, introduced by the LBM boundary conditions on the cavity's sides, results in turbulence intensities not being as low as the experimental ones in the external perimeter of the jet. We should note that this effect appears to be more important for the spanwise turbulence intensities, because of the quite strong approximation in terms of the inlet spanwise velocity profile.

CHAPTER 6 CONCLUSION AND FUTURE WORKS

The objective of the present work was the step-by-step verification and validation of the LBM method. The developed LBM code was at first aiming to correctly model 2D flows in a variety of situations and was later extended to 3D and tested for turbulence simulations. Different boundary conditions were analyzed and their performances compared. Methods aiming to increase the LBM stability were also implemented and compared to the original implementation. In all of the simulated cases, excellent agreement with either the analytical, CFD or experimental results was observed.

For the Poiseuille and Couette-Poiseuille flows, LBM simulations provide results extremely close to the corresponding analytical solutions of the N-S equations. The complex flow in the lid-driven cavity is correctly reproduced and yields results extremely close to the [15] benchmark. The unsteady flow past a square obstacle in a duct was tested for various Reynolds numbers in terms of velocity profiles at different positions, drag coefficient and Strouhal numbers. In all of these cases, the parameters appear in close agreement with the literature.

The 3D LBM was first implemented for the pressure driven flow between infinite parallel plates, showing excellent agreement with the analytical solution of the N-S equations for this particular case. The pressure driven flow in a square section duct was also implemented. Excellent agreement was observed between the LBM simulation and the quasi-analytical solution, despite the fact that this latter is obtained for a quasi-square section duct. The 3D LBM with LES turbulence model was found to correctly reconstruct the turbulent square jet flow in a square cavity. LBM results in terms of velocity profiles and turbulence intensities in the area close to the jet exit are in good agreement with the experimental work of [35].

It was therefore concluded that the LBM is a valid tool for the simulation of 2D and 3D laminar and turbulent flows.

It was found that the MRT collision model is able to greatly reduce, if not eliminate entirely, the stability related issues present in the BGK-SRT collision implementation. In particular, in the complex lid-driven cavity flow, the BGK-SRT collision model was found to become unstable and diverge for Reynolds numbers higher than 2000. In 3D, the BGK-SRT collision generates unwanted fluctuations in the velocity profiles which are not present in the experimental results nor in the MRT collision based implementation.

The implemented boundary conditions in all laminar, 2D or 3D, steady or unsteady flows,

do not present stability issues and enable to reach excellent results in terms of accuracy. The Zou-He condition is characterized by second order accuracy, thus enabling to maintain the original second order accuracy of the LBM scheme. The bounce-back boundary condition is, on the other hand, of first order in terms of accuracy. Despite this flaw, it is often preferred in simulations thanks to its easy implementation which is substantially the same in 2D and 3D.

The main area which would require further study is the turbulence implementation in the LBM. In particular the fluid-wall interactions in turbulent flows were not tested in the present work and could be the object of further studies. As an example, the performance of a refined mesh closer to the walls, or in other areas of interest, would be interesting to analyze. It should be also noted that the most widespread turbulence model implementations in the LBM are limited to DNS, LES or rely on the dissipation of the numerical scheme itself to model the viscous dissipation of the smaller scales. DNS could become a valid simulation tool in the years to come, provided the progress in terms of calculator's performance maintains its recent pace. For the time being, however, turbulence simulations should include a turbulence model and it would be interesting to implement other turbulence models in the LBM. A further challenge would be the implementation of the MRT collision and LES turbulence model with other lattice models in order to verify whether a different discretization of the velocity space would impact positively the result's accuracy.

REFERENCES

- [1] D. J. Bespalko. *Validation of the Lattice Boltzmann Method for Direct Numerical Simulation of Wall-Bounded Turbulent Flows*. PhD thesis, Queen's University, 2011.
- [2] P. L. Bhatnagar, E. P. Gross, and M. Krook. A model for collision processes in gases. *Physical review*, 94(3):511, 1954.
- [3] M. Bouzidi, D. D'Humieres, P. Lallemand, and L.-S. Luo. Lattice boltzmann equation on a 2d rectangular grid. Technical report, DTIC Document, 2002.
- [4] M. Bouzidi, M. Firdaouss, and P. Lallemand. Momentum transfer of a boltzmann-lattice fluid with boundaries. *Physics of Fluids (1994-present)*, 13(11):3452–3459, 2001.
- [5] M. Breuer, J. Bernsdorf, T. Zeiser, and F. Durst. Accurate computations of the laminar flow past a square cylinder based on two different methods: lattice-boltzmann and finite-volume. *International Journal of Heat and Fluid Flow*, 21(2):186–196, 2000.
- [6] J. E. Broadwell. Shock structure in a simple discrete velocity gas. *Physics of Fluids (1958-1988)*, 7(8):1243–1247, 1964.
- [7] V. Canuto and Y. Cheng. Determination of the smagorinsky-lilly constant cs . *Physics of Fluids*, 9:1368–1378, 1997.
- [8] H. Chen, S. Chen, and W. H. Matthaeus. Recovery of the navier-stokes equations using a lattice-gas boltzmann method. *Physical Review A*, 45(8):R5339, 1992.
- [9] N. Chen and H. Yu. Mechanism of axis switching in low aspect-ratio rectangular jets. *Computers & Mathematics with Applications*, 67(2):437–444, 2014.
- [10] S. Chen and G. D. Doolen. Lattice boltzmann method for fluid flows. *Annual review of fluid mechanics*, 30(1):329–364, 1998.
- [11] D. d'Humieres. Generalized lattice-boltzmann equations. *Rarefied gas dynamics- Theory and simulations*, 3:450–458, 1994.
- [12] D. d'Humières. Multiple-relaxation-time lattice boltzmann models in three dimensions. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 360(1792):437–451, 2002.

- [13] R. Franke, W. Rodi, and B. Schöning. Numerical calculation of laminar vortex-shedding flow past cylinders. *Journal of Wind Engineering and Industrial Aerodynamics*, 35:237–257, 1990.
- [14] S. Geller, S. Uphoff, and M. Krafczyk. Turbulent jet computations based on mrt and cascaded lattice boltzmann models. *Computers & Mathematics with Applications*, 65(12):1956–1966, 2013.
- [15] U. Ghia, K. N. Ghia, and C. Shin. High- Re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of computational physics*, 48(3):387–411, 1982.
- [16] X. He and G. Doolen. Lattice boltzmann method on curvilinear coordinates system: flow around a circular cylinder. *Journal of Computational Physics*, 134(2):306–315, 1997.
- [17] X. He and L.-S. Luo. Lattice boltzmann model for the incompressible navier-stokes equation. *Journal of statistical Physics*, 88(3-4):927–944, 1997.
- [18] S. Hou, J. Sterling, S. Chen, and G. Doolen. A lattice boltzmann subgrid model for high reynolds number flows. *Pattern Formation and Lattice Gas Automata*, 6:149, 1996.
- [19] S. Hou, Q. Zou, S. Chen, G. Doolen, and A. C. Cogley. Simulation of cavity flow by the lattice boltzmann method. *Journal of computational physics*, 118(2):329–347, 1995.
- [20] L. Jahanshaloo, E. Pouryazdanpanah, and N. A. Che Sidik. A review on the application of the lattice boltzmann method for turbulent flow simulation. *Numerical Heat Transfer, Part A: Applications*, 64(11):938–953, 2013.
- [21] L. P. Kadanoff. On two levels. *Physics today*, 39:7, 1986.
- [22] T. Kanki and S. Iuchi. Poiseuille flow and thermal creep of a rarefied gas between parallel plates. *Physics of Fluids (1958-1988)*, 16(5):594–599, 1973.
- [23] M. Krafczyk, J. Tölke, and L.-S. Luo. Large-eddy simulations with a multiple-relaxation-time lbe model. *International Journal of Modern Physics B*, 17(1):33–39, 2003.
- [24] P. Lallemand and L.-S. Luo. Theory of the lattice boltzmann method: Dispersion, dissipation, isotropy, galilean invariance, and stability. *Physical Review E*, 61(6):6546, 2000.
- [25] J. Latt. *Hydrodynamic limit of lattice Boltzmann equations*. PhD thesis, University of Geneva, 2007.

- [26] J. Latt, B. Chopard, O. Malaspinas, M. Deville, and A. Michler. Straight velocity boundaries in the lattice boltzmann method. *Physical Review E*, 77(5):056703, 2008.
- [27] S. Leclaire. *Étude d'un modèle de Boltzmann sur réseau pour la simulation assistée par ordinateur des fluides à plusieurs phases immiscibles*. PhD thesis, École Polytechnique de Montréal, 2013.
- [28] G. R. McNamara and G. Zanetti. Use of the boltzmann equation to simulate lattice-gas automata. *Physical Review Letters*, 61(20):2332, 1988.
- [29] W. Miller. Flow in the driven cavity calculated by the lattice boltzmann method. *Physical Review E*, 51(4):3659, 1995.
- [30] D. R. Noble, J. G. Georgiadis, and R. O. Buckius. Comparison of accuracy and performance for lattice boltzmann and finite difference simulations of steady viscous flow. *International Journal for Numerical Methods in Fluids*, 23(1):1–18, 1996.
- [31] B. H. Y. Pomeau and U. Frisch. Lattice-gas automata for the navier-stokes equation. *Phys. Rev. Lett*, 56(14):1505, 1986.
- [32] S. B. Pope. *Turbulent flows*. Cambridge university press, 2000.
- [33] A. K. Prasad and J. R. Koseff. Reynolds number and end-wall effects on a lid-driven cavity flow. *Physics of Fluids A: Fluid Dynamics (1989-1993)*, 1(2):208–218, 1989.
- [34] Y. Qian, D. d’Humières, and P. Lallemand. Lattice bgk models for navier-stokes equation. *EPL (Europhysics Letters)*, 17(6):479, 1992.
- [35] W. Quinn and J. Militzer. Experimental and numerical study of a turbulent free square jet. *Physics of Fluids (1958-1988)*, 31(5):1017–1025, 1988.
- [36] P. J. Roache. Quantification of uncertainty in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 29(1):123–160, 1997.
- [37] J. Smagorinsky. General circulation experiments with the primitive equations: the basic experiment. *Monthly weather review*, 91(3):99–164, 1963.
- [38] sourceforge.net. *Plot Digitizer*, 2014.
- [39] S. Succi. *The lattice Boltzmann equation: for fluid dynamics and beyond*. Oxford university press, 2001.

- [40] S. Succi, E. Foti, and F. Higuera. Three-dimensional flows in complex geometries with the lattice boltzmann method. *EPL (Europhysics Letters)*, 10(5):433, 1989.
- [41] M. Sukop and D. Thorne. *Lattice Boltzmann Modeling*. Springer, 2006.
- [42] A. Tamayol and M. Bahrami. Laminar flow in microchannels with noncircular cross section. *Journal of Fluids Engineering*, 132(11):111201, 2010.
- [43] S. Wolfram et al. *Theory and applications of cellular automata*, volume 1. World Scientific Singapore, 1986.
- [44] H. Yu. *Lattice Boltzmann equation simulations of turbulence, mixing, and combustion*. PhD thesis, Texas A&M University, 2004.
- [45] H. Yu, L.-S. Luo, and S. S. Girimaji. Les of turbulent square jet flow using an mrt lattice boltzmann model. *Computers & Fluids*, 35(8):957–965, 2006.
- [46] D. P. Ziegler. Boundary conditions for lattice boltzmann simulations. *Journal of Statistical Physics*, 71(5-6):1171–1177, 1993.
- [47] Q. Zou and X. He. On pressure and velocity boundary conditions for the lattice boltzmann bgk model. *Physics of Fluids (1994-present)*, 9(6):1591–1598, 1997.

APPENDIX A Lattice Comparisons in 3D

For the D3Q15 lattice, discrete velocities are:

$$e_i = \begin{cases} (0, 0, 0) & i = 1 \\ (\pm 1, 0, 0), \quad (0, \pm 1, 0), \quad (0, 0, \pm 1) & i = 2, 3, \dots, 7 \\ (\pm 1, \pm 1, \pm 1) & i = 8, 9, \dots, 15 \end{cases} \quad (\text{A.1})$$

The weight coefficients for the D3Q15 lattice are:

$$w_i = \begin{cases} 2/9 & i = 1 \\ 1/9 & i = 2, 3, \dots, 7 \\ 1/72 & i = 8, 9, \dots, 19 \end{cases} \quad (\text{A.2})$$

For the D3Q27 lattice, discrete velocities are:

$$e_i = \begin{cases} (0, 0, 0) & i = 1 \\ (\pm 1, 0, 0), \quad (0, \pm 1, 0), \quad (0, 0, \pm 1) & i = 2, 3, 4, 5, 18, 19 \\ (\pm 1, \pm 1, 0), \quad (\pm 1, 0, \pm 1), \quad (0, \pm 1, \pm 1) & i = 6, 7, \dots, 17 \\ (\pm 1, \pm 1, \pm 1) & i = 20, 21, \dots, 27 \end{cases} \quad (\text{A.3})$$

The weight coefficients for the D3Q27 lattice are:

$$w_i = \begin{cases} 8/27 & i = 1 \\ 2/27 & i = 2, 3, \dots, 7 \\ 1/54 & i = 8, 9, \dots, 19 \\ 1/216 & i = 20, 21, \dots, 27 \end{cases} \quad (\text{A.4})$$

Simulations are run following the same criteria for mesh, reference velocity, Reynolds number and convergence indicated in Sections 5.2.1 and 5.2.2 which describe the pressure driven test cases in detail.

Table A.1 Lattice comparisons

Lattice Type	Mesh	Collision	Test Case	Error
D3Q15	3x61x61	BGK-SRT	Parallel infinite plates	$2.868 \cdot 10^{-4}$
D3Q19	3x61x61	BGK-SRT	Parallel infinite plates	$2.868 \cdot 10^{-4}$
D3Q27	3x61x61	BGK-SRT	Parallel infinite plates	$2.868 \cdot 10^{-4}$
D3Q15	3x61x61	BGK-SRT	Channel Flow	$6.096 \cdot 10^{-3}$
D3Q19	3x61x61	BGK-SRT	Channel Flow	$2.927 \cdot 10^{-3}$
D3Q27	3x61x61	BGK-SRT	Channel Flow	$2.106 \cdot 10^{-3}$

We notice that, in the simulation of a flow between flat, parallel, infinite plates, the three lattices yield very similar results. This test case is therefore not exigent enough to enable any differentiation between the three lattices. The simulation of a pressure driven flow in a square section duct, on the other hand, shows that the D3Q19 lattice provides better results compared to the D3Q15 and the D3Q27 lattices. Difference between results appear however too small to reach definitive conclusions on the three lattice's performances.

APPENDIX B MRT matrix for D3Q19 cubic lattice

The M matrix, responsible for the switching from the velocity to the moment space, has the following expression for the D3Q19 cubic lattice:

$$\left\{ \begin{array}{l}
 M = (\varphi_1, \varphi_2, \varphi_3, \dots, \varphi_{17}, \varphi_{18}, \varphi_{19}) \\
 \varphi_1 = \sqrt{e_x^2 + e_y^2 + e_z^2}^0 \\
 \varphi_2 = 19 \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 - 30 \right) \\
 \varphi_3 = 0.5 \left[21 \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^4 - 53 \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 + 24 \right) \right] \\
 \varphi_4 = e_x \\
 \varphi_5 = e_x \left[5 \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 - 9 \right) \right] \\
 \varphi_6 = e_y \\
 \varphi_7 = e_y \left[5 \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 - 9 \right) \right] \\
 \varphi_8 = e_z \\
 \varphi_9 = e_z \left[5 \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 - 9 \right) \right] \\
 \varphi_{10} = 3e_x^2 - \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 \right)^2 \\
 \varphi_{11} = \left[3 \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 - 5 \right) \right] \left[3e_x^2 - \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 \right)^2 \right] \\
 \varphi_{12} = e_y^2 - e_z^2 \\
 \varphi_{13} = \left[3 \left(\sqrt{e_x^2 + e_y^2 + e_z^2}^2 - 5 \right) \right] (e_y^2 - e_z^2) \\
 \varphi_{14} = e_x \cdot e_y \\
 \varphi_{15} = e_y \cdot e_z \\
 \varphi_{16} = e_x \cdot e_z \\
 \varphi_{17} = (e_y^2 - e_z^2) e_x \\
 \varphi_{18} = (e_z^2 - e_x^2) e_y \\
 \varphi_{19} = (e_x^2 - e_y^2) e_z
 \end{array} \right. \tag{B.1}$$

APPENDIX C Square Duct Flow: Quasi-Analytical Solution

The procedure suggested by [42] is valid for hyperelliptical and regular polygonal cross section ducts.

Prior to presenting their analysis, we have to consider what is an hyperellipse. If we define a and b as the major and minor axes of the cross-section and ϵ as their ratio $\epsilon = \frac{b}{a}$, in the first quadrant of a Cartesian plane the boundary of an hyperellipse is described by:

$$r_0 = \frac{a}{\left((\cos \theta)^n + \left(\frac{\sin \theta}{\epsilon} \right)^n \right)^{\frac{1}{n}}} \quad (\text{C.1})$$

This equation describes only the boundary (r_0) of the hyperellipse and by varying the n parameter we can obtain several geometries, such as the ones shown in Figure C.1.

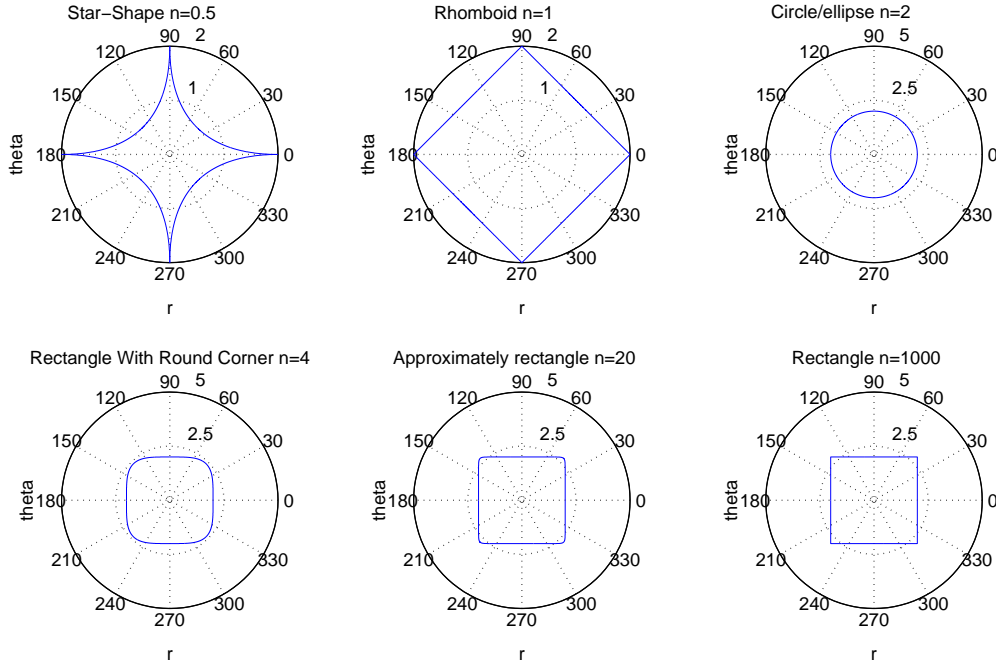


Figure C.1 Geometries obtainable for variable n with unitary aspect ratio ϵ

As stated previously the boundary equation is defined only in the first quadrant, but the results can be extended to the other three through symmetry.

The Navier-Stokes equations in polar coordinates for an incompressible, fully developed, laminar, pressure driven flow in a constant hyperelliptical section fall down to the Poisson equation:

$$\frac{dp}{dz} = \mu \left(\frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} + \frac{1}{r^2} \frac{\partial^2 u}{\partial \theta^2} \right) \quad (\text{C.2})$$

where r is the radial coordinate and varies from 0 to r_0 and the angle θ is related to the first quadrant and therefore is included in the 0 to $\pi/2$ interval.

In the first quadrant, we can impose three boundary conditions:

$$\begin{cases} \frac{\partial u}{\partial \theta}_{\theta=\pi/2} = 0 \\ \frac{\partial u}{\partial \theta}_{\theta=0} = 0 \\ u(r_0) = 0 \end{cases} \quad (\text{C.3})$$

Equation C.2 can be solved and nondimensionalized, and becomes:

$$\begin{cases} u^* = \frac{u}{u_{max}} = 1 - \frac{1}{4A_1} \eta^2 + \sum_{i=1}^{\infty} \frac{C_i}{A_1} \eta^{2i} \cos 2i\theta \\ u_{max} = \frac{1}{\mu \left(\frac{dP}{dz} A_1 a^2 \right)} \end{cases} \quad (\text{C.4})$$

where η is the non-dimensional coordinate equal to r/a . Applying the boundary conditions, the unknown coefficients A_1 and C_i can be calculated.

To obtain the analytical solution for almost square section ducts, these coefficients were found by [42] for the maximum reported value of $n=20$ and unitary aspect ratio ϵ :

$$A_1 = 0.296 \quad C_1 = 0 \quad C_2 = -0.046 \quad C_3 = C_4 = C_5 = C_{\infty} = 0$$

APPENDIX D CPU Time Measurements

The LBM codes for the various test cases were all implemented using MATLAB R2013[®] run on a Windows 7 based computer characterized by the following processor: INTEL CORE i7 @ 3.20 GHz with a RAM of 16 GB.

Great effort was made to use the same implementation techniques and achieve equal vectorization on all of the different test cases.

Table D.1 provides the running time of all of the LBM codes implemented for the present work. Indications concerning Reynolds number, mesh density, collision model and total iterations required for convergence are provided in the table. For a more comprehensive presentation of the test cases with more detailed information, the reader is referred to the corresponding Sections in the main document.

Table D.1 CPU Time Measurements for the various test cases

Test Case	Re	Mesh	Collis.	Total Iter.	Time for 1000 It [s]	Total Time
2D Poi (bounce-back)	400	4x101	SRT	1184000	0.7665*	14.7 min
2D Poi (bounce-back)	400	4x101	MRT	1184000	0.6475*	12.38 min
2D Poi (Zou-He)	400	4x101	SRT	1160000	0.7760*	15.1 min
2D Poi (Zou-He)	400	4x101	MRT	1160000	0.6320*	12.2 min
2D Cou-Poi	300	3x61	SRT	452000	0.6832*	5.14 min
2D Cou-Poi	300	3x61	MRT	452000	0.4372*	3.27 min
2D Cavity	400	257x257	SRT	256000	16.32	69.6 min
2D Cavity	400	257x257	MRT	256000	19.15	81.7 min
2D Obstacle	100	2000x320	SRT	91122	197.9	5.01 hrs
3D Flat Plates	500	3x61x61	SRT	736000	5.327	65.4 min
3D Flat Plates	500	3x61x61	MRT	736000	6.957	85.3 min
3D Channel	500	3x61x61	SRT	526000	5.417	47.5 min
3D Channel	500	3x61x61	MRT	526000	6.912	60.6 min
3D Turbo Jet	184000	600x121x121	SRT	47957	12323	6.84 days
3D Turbo Jet	184000	600x121x121	MRT	47957	5574	3.09 days

Times labeled with a (*) were, in fact, calculated for a more significant number of iterations, since their corresponding running time for 1000 iterations was very short and hence subject

to a large error of measurement.

In general, we expect the running time to be greatly dependent on the mesh density. The Reynolds also has an influence since it is used to determine the viscosity and hence affects the convergence of the solution. No significant difference should be observed, in terms of calculation time, between the Zou-He and bounce-back boundary conditions. We would expect the MRT collision model to be more computationally intensive with respect to the BGK-SRT collision model, because of the extra calculations required for the switching between the velocity and moment based spaces.

We notice that these observations are generally supported by results presented in Table D.1. Differences between what is expected and what is actually observed appear for the 2D Poiseuille and Couette-Poiseuille test cases where the running times for the MRT collision are lower than the BGK-SRT collision. This can be explained by the fact that the MRT collision can be expressed in a more vectorized way with respect to the BGK-SRT collision. This yields an advantage for the coarse meshes associated to the pressure driven 2D flows. As the meshes get finer, the larger number of calculations required for the MRT outweighs this advantage and it can be seen that running times for MRT collisions are higher than the BGK-SRT collision. The only exception to this is the turbulent jet simulation. We should however consider that, in this case, the implementation of the LES turbulence model influences the running time. In particular, its implementation in a BGK-SRT collision is far more computationally intensive than the LES-MRT based. This results in calculation times for the BGK-SRT collision code being larger.